REMEDI: EDITABLE CONCEPT MEMORY FOR FROZEN DENSE RETRIEVERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Dual-encoder retrievers often fail on short queries that name concepts for which the query encoder lacks parametric knowledge. Prior work typically addresses this problem by fine-tuning on in-domain data, but doing so can degrade zero-shot retrieval on other tasks. We introduce REMEDI (**R**etrieval **E**nrichment via **M**emory-**E**ditable **D**ictionary **I**nterfaces), which keeps the retriever frozen and instead augments queries with an external, user-editable concept memory: a dictionary mapping each concept to a small set of textual descriptions. REMEDI encodes these descriptions once using the frozen document encoder and trains only a lightweight cross-attention adapter, leaving the pretrained encoder parameters, document embeddings, and retrieval index unchanged. At test time, REMEDI matches query spans to concepts, attends over the corresponding descriptions to construct a memory token, and inserts this token into the query between learned boundary tokens. Across eight benchmarks and five encoders, REMEDI achieves the best average nDCG@10 on every encoder, outperforming full fine-tuning and LoRA, with the largest gains on concept-heavy queries. It also generalizes to concepts unseen during training and stays within 0.64 points of the frozen retriever on out-of-domain BEIR tasks, where fine-tuning loses up to 5.2. Theoretically, we characterize when relevant memory lifts a query's positive above its hard negatives, and show when minimizing the contrastive loss guarantees correct ranking, first for linear encoders and then for Transformers.

1 INTRODUCTION

Dual encoders form the workhorse of modern large-scale retrieval systems (Karpukhin et al., 2020; Reimers & Gurevych, 2019; Izacard et al., 2022). They map queries and documents into a shared space, where relevance reduces to vector similarity. Because document vectors are computed offline and indexed, retrieval requires only one query-encoder pass and one lookup. Pretraining on large, diverse corpora makes modern embedding models broadly effective zero-shot across domains and tasks (Thakur et al., 2021; Wang et al., 2022b; Ni et al., 2022), and the same machinery underpins retrieval-augmented generation (RAG) (Lewis et al., 2020).

This strength has a blind spot. A dual encoder builds a query's representation from two ingredients: the tokens in the query and the parametric knowledge it absorbed during pretraining. The second ingredient runs dry when a query hinges on a specialized concept that appeared rarely, or not at all, in the pretraining data (Kandpal et al., 2023; Mallen et al., 2023). Consider the query *"What is (are) Dystonias?"* from the MedQuAD dataset (Ben Abacha & Demner-Fushman, 2019), whose positive document explains that dystonias involve sustained muscle contractions, twisting movements, and abnormal postures. If the encoder has not learned *Dystonias*, it cannot place this query near that document. The example exposes an **information asymmetry** between the two towers of a dual encoder: even when they share parameters, the document can ground an unfamiliar concept in its surrounding text, whereas the query names it in a few words and must lean on whatever its parameters already encode.

Prior work predominantly addresses this gap through fine-tuning on in-domain data, but such approaches trade one problem for another: the encoder must absorb new concepts without eroding the broad zero-shot ability that made it useful (Ovadia et al., 2024; Wang et al., 2022a). LoRA (Hu et al., 2022), adapters (Houlsby et al., 2019), and regularizers such as elastic-weight consolidation

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

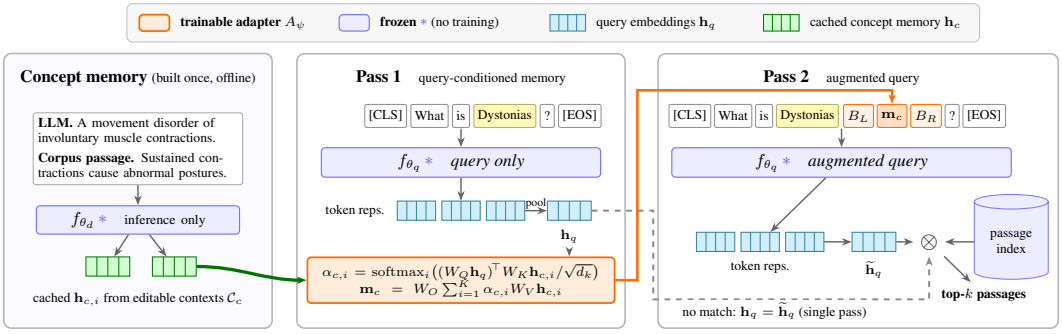


Figure 1: **REMEDI augments a frozen dual encoder with editable concept memory.** *Left:* concept contexts are encoded once and cached. *Center:* the pooled query $\mathbf{h}_q$ attends over cached memory to form $\mathbf{m}_c$. *Right:* $\mathbf{m}_c$ is inserted between B_L and B_R; the re-encoded query searches the unchanged index. Queries without concepts skip memory ($\mathbf{h}_q = \tilde{\mathbf{h}}_q$).

(Kirkpatrick et al., 2017) can effectively constrain parameter updates, but cannot guarantee that the original retrieval behavior is preserved as-is. Indeed, our experiments show that even modest parameter updates can hurt performance on held-out BEIR tasks. This motivates our central question: *can a deployed retriever learn new in-domain concepts without changing its encoder or sacrificing out-of-domain zero-shot performance?*

Enriching the query is a natural remedy. Pseudo-relevance feedback (PRF), a popular approach, expands the query with the top results of an initial search (Yu et al., 2021; Li et al., 2025). When that search fails, however, PRF reinforces the error. We instead draw evidence from outside the retrieval loop. We propose REMEDI (Figure 1), which enriches queries from an external *concept memory*: a user-editable dictionary that maps each concept to a few textual descriptions, written by an LLM or retrieved from the corpus with BM25 (Robertson & Zaragoza, 2009). The frozen document encoder embeds each description once. A lightweight cross-attention adapter, the only trained component, condenses the memory of each matched concept into a memory token, which is inserted after the matched span; the frozen encoder then re-encodes the enriched query and searches the index.

The adapter learns *how* to use knowledge, while the dictionary holds *what* is known. Hence, knowledge stays editable, even for concepts unseen in training, without retraining; queries that match no concept take the original retrieval path, preserving out-of-domain behavior by construction; and, unlike generative expansion (Gao et al., 2023; Wang et al., 2023), no LLM is called at query time. We prove when memory lifts a query above its hard negatives and when a low contrastive loss certifies the ranking. Across eight datasets and five encoders, REMEDI achieves the best average nDCG@10 on every encoder (65.9 vs. 63.5 for full fine-tuning), stays within 0.64 points of the frozen model on BEIR, where fine-tuning loses up to 5.2, and improves retrieval on unseen concepts.

2 Motivation: The Query Does Not Tell the Whole Story

In MedQuAD (Ben Abacha & Demner-Fushman, 2019), each short consumer-health question has one gold passage among 16,393 passages; the query often only *names* a concept that its answer *explains*. Figure 2 traces two such queries through a frozen retriever and through REMEDI, which share the same encoder and index; GPT-4o labels each top-10 result by whether it matches the query's concept, its question, both, or neither. **In the top row, the concept is unknown.** Frozen RoBERTa returns ten unrelated passages and ranks the gold definition beyond 100. REMEDI places 0.78 of its attention on the definition slot, and the gold rises to rank 1. **In the bottom row, the answer facet is missed.** Qwen3 recognizes the disorder but retrieves the *right concept* answering the *wrong question*, leaving the gold at rank 85. REMEDI injects the *ATP1A3* evidence from memory, and the gold moves to rank 1. In both cases, REMEDI supplies at query time the knowledge the query tower's parameters fail to encode.

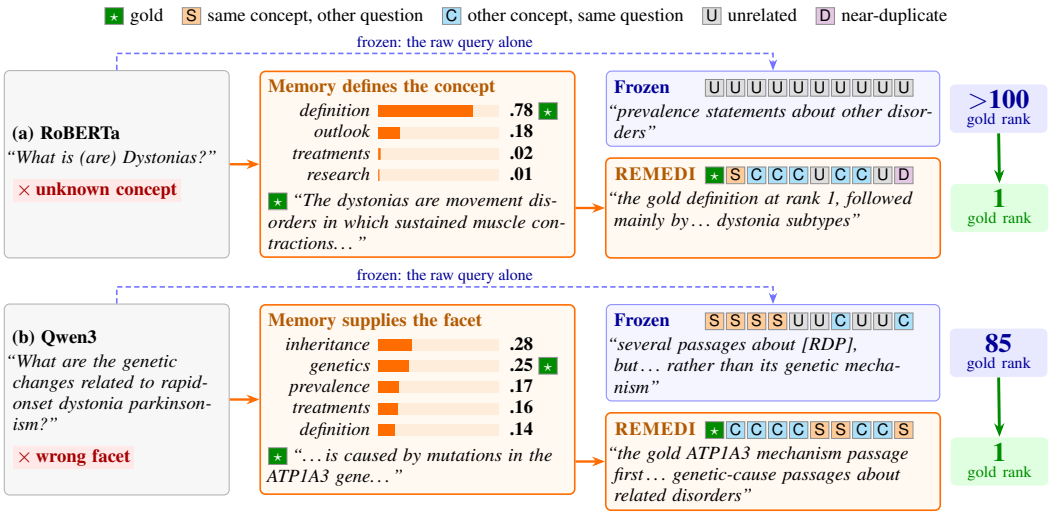


Figure 2: Two frozen-retriever failures, one repair. Rows read left to right: query, concept memory with REMEDI's attention, top-10 rankings, and gold rank. Labels and quotes: blinded GPT-4o.

3 Related Work

Dense retrievers generalize poorly to rare and unseen entities (Kandpal et al., 2023; Mallen et al., 2023; Sciavolino et al., 2021). *Parametric adaptation* (full fine-tuning, adapters, LoRA; Houlsby et al., 2019; Hu et al., 2022) writes new knowledge into the query tower and risks out-of-domain drift. *Query enrichment* either trusts a first-pass ranking (Yu et al., 2021; Li et al., 2023), which fails when a concept is unfamiliar, or calls an LLM per query (Gao et al., 2023; Wang et al., 2023). *Memory-augmented retrievers* need a second index (MoMA; Ge et al., 2023) or learned entity embeddings that cannot represent unseen concepts (KPR; Yamada et al., 2025). REMEDI keeps the encoder and index frozen and reads user-editable memory, so adding a concept requires no retraining. Appendix A discusses related work in detail.

4 Problem Formulation

Let $\mathcal{Q}$ denote the query space and $\mathcal{D}$ the document corpus. We start with a pretrained dual encoder $f_\theta = (f_{\theta_q}, f_{\theta_d})$, where the query and document towers map their inputs to $\mathbb{R}^h$. In many settings, the two towers share parameters. Before deployment, the system encodes each document once as $\mathbf{z}_d = f_{\theta_d}(d)$ for $d \in \mathcal{D}$, and stores the resulting embeddings in a retrieval index. Given a query $q \in \mathcal{Q}$, the query tower produces $\mathbf{z}_q = f_{\theta_q}(q)$ and ranks documents by cosine similarity $s_\theta(q, d) = \mathrm{sim}(\mathbf{z}_q, \mathbf{z}_d)$. We consider the deployment setting in which the document tower f_{θ_d} and the index $\{\mathbf{z}_d\}_{d\in\mathcal{D}}$ are fixed. Consequently, any adaptation method must improve query representations without re-encoding the corpus or changing the document-side parameters.

In addition to the retriever, we assume access to an external memory realized through a concept dictionary $\mathcal{M} = \{c \mapsto \mathcal{C}_c\}$ that, for each concept c, provides a small set of textual contexts $\mathcal{C}_c = \{x_{c,1}, \ldots, x_{c,K}\}$ that describe it. A concept may be an entity, phrase, subject, or any user-defined textual span. Its contexts may be drawn from $\mathcal{D}$, generated by prompting an LLM, or collected from external resources such as Wikipedia. For a tokenized query $q = (t_1, \ldots, t_L)$, a segment matcher identifies spans that match concepts in $\mathcal{M}$. Let $c_1, \ldots, c_{m(q)}$ denote the matched concepts in their order of occurrence, where c_j matches $(t_{a_j}, \ldots, t_{b_j})$. The query therefore activates only the corresponding memory entries $\mathcal{M}(q) = \{(c_j, \mathcal{C}_{c_j})\}_{j=1}^{m(q)}$.

Our goal in this paper is to learn a lightweight query-side adapter A_ψ that uses $\mathcal{M}(q)$ to construct an improved query representation while preserving compatibility with the fixed document index. The adapter must leave the retriever unchanged when no concept matches the query. The next section describes our memory interface and how we train it under these constraints.

5 SOLUTION APPROACH

REMEDI realizes the adaptation through A_ψ, a lightweight cross-attention module between the query and its matched memory entries $\mathcal{M}(q)$. A_ψ selects relevant information from the contexts associated with each matched concept and injects it into the query input. For every context $x_{c,i} \in \mathcal{C}_c$, we compute its embedding once with the frozen document tower, $\mathbf{h}_{c,i} = f_{\theta_d}(x_{c,i})$, and cache $\{\mathbf{h}_{c,i}\}_{i=1}^K$ with the corresponding dictionary entry. For a query q with at least one matched concept, we first run the frozen query tower to obtain a query representation $\mathbf{h}_q = f_{\theta_q}(q)$. Then, for a concept c, we project the query representation and its cached context embeddings into a shared attention space, $\mathbf{r}_q = W_Q\mathbf{h}_q$, $\mathbf{k}_{c,i} = W_K\mathbf{h}_{c,i}$, and $\mathbf{v}_{c,i} = W_V\mathbf{h}_{c,i}$, and attend only over the K contexts associated with c through cross-attention. The attention weights are $\alpha_{c,i} = \mathrm{softmax}_i(\mathbf{r}_q^\top \mathbf{k}_{c,i}/\sqrt{d_k})$ and the pooled memory is $\mathbf{u}_c = \sum_{i=1}^K \alpha_{c,i}\mathbf{v}_{c,i}$. An output projection maps this to the input-embedding dimension of the query tower, $\mathbf{m}_c = W_O\mathbf{u}_c$, yielding a continuous memory token.

Let $\mathbf{E}_{\theta_q}(q) = (\mathbf{e}_1, \ldots, \mathbf{e}_L)$ denote the original query-token embeddings. For a matched concept c_j spanning tokens a_j through b_j, REMEDI preserves the original span and inserts its memory token immediately after it. Two learned boundary embeddings, $\mathbf{B}_\mathrm{L}$ and $\mathbf{B}_\mathrm{R}$, mark the inserted memory:

$$[\ldots; \mathbf{e}_{a_j}; \ldots; \mathbf{e}_{b_j}; \mathbf{B}_\mathrm{L}; \mathbf{m}_{c_j}; \mathbf{B}_\mathrm{R}; \mathbf{e}_{b_j+1}; \ldots]. \tag{1}$$

Preserving the matched tokens retains the query's original lexical evidence, while the boundary embeddings provide a learned interface through which the frozen encoder can interpret the inserted memory. When several concepts match a query, we insert all corresponding memory tokens in their order of occurrence. The frozen query tower then encodes the resulting augmented sequence as $\widetilde{\mathbf{z}}_q = f_{\theta_q}(\widetilde{\mathbf{E}}_\psi(q))$. For queries with no matched concept, REMEDI bypasses the interface and uses the original query representation.

We train the adapter on queries with at least one dictionary match. Given a positive document d^+ and a set of mined negatives $\mathcal{N}(q)$, we optimize A_ψ using the standard contrastive objective:

$$\mathcal{L}_{\mathrm{InfoNCE}}(q) = -\log \frac{\exp\big(\widetilde{\mathbf{z}}_q^\top f_{\theta_d}(d^+)/\tau\big)}{\exp\big(\widetilde{\mathbf{z}}_q^\top f_{\theta_d}(d^+)/\tau\big) + \sum_{d^-\in\mathcal{N}(q)} \exp\big(\widetilde{\mathbf{z}}_q^\top f_{\theta_d}(d^-)/\tau\big)}, \tag{2}$$

where $\tau > 0$ is a temperature parameter. The trainable parameters are therefore a set of linear projection matrices and a pair of boundary tokens, $\psi = \{W_Q, W_K, W_V, W_O, \mathbf{B}_\mathrm{L}, \mathbf{B}_\mathrm{R}\}$.

6 WHEN DOES CONCEPT MEMORY IMPROVE A FROZEN RETRIEVER?

We analyze when concept memory $\mathcal{M}$ improves a frozen retriever. We first model f_{θ_q} as linear, which abstracts away the nonlinear, position-sensitive operations of a Transformer and isolates the effect of memory selection analytically; we then extend the main guarantee to Transformers. Appendix B contains all proofs.

Linear model. For query-token embeddings $\mathbf{E}_{\theta_q}(q) = (\mathbf{e}_1, \ldots, \mathbf{e}_L)$, the linear query tower is $\mathbf{g}_{\theta_q}(q) = W_f \sum_{t=1}^L \mathbf{e}_t$ and $f_{\theta_q}(q) = \mathbf{g}_{\theta_q}(q)/\|\mathbf{g}_{\theta_q}(q)\|_2$. Given a positive d^+ and $R \geq 1$ negatives $d_1^-, \ldots, d_R^-$, write $\mathbf{u}_q = \mathbf{g}_{\theta_q}(q)$, $A = \|\mathbf{u}_q\|_2 > 0$, and $\mathbf{z}_q = \mathbf{u}_q/A$ for the frozen query, and $\mathbf{z}^+ = f_{\theta_d}(d^+)$, $\mathbf{z}_j^- = f_{\theta_d}(d_j^-)$ for the documents.

Memory augmentation by A_ψ. Suppose the query contains a single matched concept c with cached context embeddings $\mathbf{h}_k = \mathbf{h}_{c,k}$, $k = 1, \ldots, K$. A_ψ forms the query $\mathbf{r}_q = W_Q\mathbf{z}_q$, the keys $\mathbf{k}_k = W_K\mathbf{h}_k$, and the values $W_V\mathbf{h}_k$. It scores context k by the logit $\ell_k = \mathbf{r}_q^\top\mathbf{k}_k$, gives it attention weight $\alpha_k = \mathrm{softmax}_k(\ell_k/\tau_{\mathrm{att}})$ with $\tau_{\mathrm{att}} = \sqrt{d_k}$, and emits the memory token $\mathbf{m}_c = W_O\sum_{k=1}^K \alpha_k W_V\mathbf{h}_k$. The augmented query is $\widetilde{\mathbf{z}}_q = f_{\theta_q}(\widetilde{\mathbf{E}}_\psi(q))$, as defined below.

Under the linear encoder, the three inserted tokens shift $\mathbf{u}_q$ by $W_f(\mathbf{B}_\mathrm{L}+\mathbf{m}_c+\mathbf{B}_\mathrm{R})$. Since $\sum_k \alpha_k = 1$, this shift equals $\sum_k \alpha_k\mathbf{v}_k$, where $\mathbf{v}_k = W_f(\mathbf{B}_\mathrm{L} + \mathbf{B}_\mathrm{R}) + W_fW_OW_V\mathbf{h}_k$ is the shift if context k received all attention. Let $C = \max_k \|\mathbf{v}_k\|_2$. For $C > 0$, define $\widehat{\mathbf{v}}_k = \mathbf{v}_k/C$, the attended memory direction $\overline{\mathbf{v}} = \sum_k \alpha_k\widehat{\mathbf{v}}_k$, and the memory weight $\lambda = C/(A + C)$.

Lemma 6.1 (Augmented query as an interpolation)**.** *If $C > 0$, then $\|\overline{\mathbf{v}}\|_2 \le 1$, $\lambda \in (0,1)$, and*

$$\widetilde{\mathbf{z}}_q = \frac{(1-\lambda)\mathbf{z}_q + \lambda\overline{\mathbf{v}}}{\|(1-\lambda)\mathbf{z}_q + \lambda\overline{\mathbf{v}}\|_2} \tag{3}$$

whenever the numerator is nonzero. If $C = 0$, then $\widetilde{\mathbf{z}}_q = \mathbf{z}_q$.

The augmented query thus renormalizes a convex combination of $\mathbf{z}_q$ and $\overline{\mathbf{v}}$. Let $a = \mathbf{z}_q^\top \mathbf{z}^+$ and $b = \overline{\mathbf{v}}^\top \mathbf{z}^+$. If memory aligns better with d^+ than the query ($b > a$), positive similarity rises:

Lemma 6.2. *If $C > 0$ and $0 \le a < b$, then the augmented query strictly improves positive similarity: $\widetilde{\mathbf{z}}_q^\top \mathbf{z}^+ > \mathbf{z}_q^\top \mathbf{z}^+$.*

Memory helps only if attention concentrates on a context that aligns better with d^+ than the query does. Lemmas B.1 and B.2 (Appendix B) quantify the attention mass, and the equivalent logit gap, that such a context needs.

Raising positive similarity alone is not enough, since memory can sometimes raise a hard negative's similarity even more (see Remark B.3 for a worked-out toy example). Useful memory must therefore be *discriminative*. For $\widehat{\mathbf{v}}_\star \neq \mathbf{0}$, let $\mathbf{r}_\star = \widehat{\mathbf{v}}_\star / \|\widehat{\mathbf{v}}_\star\|_2$, $s_\star^+ = \mathbf{r}_\star^\top \mathbf{z}^+$, and $s_\star^- = \max_j \mathbf{r}_\star^\top \mathbf{z}_j^-$.

Lemma 6.3. *Define the worst-case margin of context $\star$ as $\omega_\star := \min_{1\le j\le R} \widehat{\mathbf{v}}_\star^\top(\mathbf{z}^+ - \mathbf{z}_j^-)$, and call context $\star$ discriminative if $\omega_\star > 0$. Then $\omega_\star = \|\widehat{\mathbf{v}}_\star\|_2(s_\star^+ - s_\star^-)$, so $\omega_\star > 0$ iff $s_\star^+ > s_\star^-$: the context is discriminative exactly when it scores the positive above every negative.*

Lemma B.4 (Appendix B) gives a simpler sufficient condition: a context close enough to d^+ is always discriminative. Discrimination must also survive attention. Each context k has worst-case margin $\omega_k = \min_j \widehat{\mathbf{v}}_k^\top(\mathbf{z}^+ - \mathbf{z}_j^-)$; suppose $\omega_\star > 0$ while $\omega_k \ge \omega_{\min}$ for $k \neq \star$, where $\omega_{\min} \ge -2$ may be negative. The attended memory $\overline{\mathbf{v}}$ has margin $\beta_j = \overline{\mathbf{v}}^\top(\mathbf{z}^+ - \mathbf{z}_j^-)$ against negative j and worst-case margin $\beta = \min_j \beta_j$. Thus ω scores a single context and β the attended mix.

Lemma 6.4. *The attended margin satisfies $\beta \ge \alpha_\star\omega_\star + (1-\alpha_\star)\omega_{\min}$. Hence $\beta > 0$ for any $\alpha_\star > 0$ if $\omega_{\min} \ge 0$, and for $\alpha_\star > -\omega_{\min}/(\omega_\star - \omega_{\min})$ if $\omega_{\min} < 0$.*

As before, Lemma B.5 (Appendix B) converts this attention mass into a logit gap.

From margins to the InfoNCE objective. The lemmas above say when the attended memory is discriminative. Two questions still remain: **(i)** when does such memory repair a frozen query that ranks a negative higher, and **(ii)** does InfoNCE (Eq. 2) reward this repair?

Let $\gamma_j = \mathbf{z}_q^\top(\mathbf{z}^+ - \mathbf{z}_j^-)$ be the frozen query's margin against negative j, and $\rho = \max_j[-\gamma_j]_+$ its ranking deficit, i.e., how far it trails the hardest negative ($\rho = 0$ when it already ranks d^+ first). By Eq. 3, the augmented margin mixes γ_j with the memory margin β_j, and the loss becomes

$$\mathcal{L}_\lambda = \log\Big(1 + \sum_{j=1}^{R} \exp\big(-\widetilde{\mathbf{z}}_q^\top(\mathbf{z}^+ - \mathbf{z}_j^-)/\tau\big)\Big), \qquad \widetilde{\mathbf{z}}_q^\top(\mathbf{z}^+ - \mathbf{z}_j^-) = \frac{(1-\lambda)\gamma_j + \lambda\beta_j}{\|(1-\lambda)\mathbf{z}_q + \lambda\overline{\mathbf{v}}\|_2}. \tag{4}$$

This loss certifies ranking: each summand is at most $\exp(\mathcal{L}_\lambda) - 1$, so $\mathcal{L}_\lambda < \log 2$ forces every margin to be positive (Appendix B.4). That argument runs backward, from an achieved loss to a ranking. The next theorem runs forward, from memory quality to a memory weight λ that suffices for repair.

Theorem 6.5 (Sufficient interpolation weight for frozen-query repair)**.** *Suppose the memory is discriminative, $\beta > 0$, and let $\mu_\lambda = \lambda\beta - (1-\lambda)\rho$. If $\lambda > \rho/(\rho+\beta)$, then $\widetilde{\mathbf{z}}_q^\top(\mathbf{z}^+ - \mathbf{z}_j^-) \ge \mu_\lambda > 0$ for all j, so $\widetilde{\mathbf{z}}_q$ ranks d^+ above every negative.*

The frozen query reduces each margin by at most $(1-\lambda)\rho$, whereas memory increases it by at least $\lambda\beta$. When $\mu_\lambda > 0$, memory overcomes the deficit and places d^+ first.

Corollary 6.6 (InfoNCE after frozen-query repair)**.** *Under Theorem 6.5, if $\lambda > \rho/(\rho+\beta)$, then $\mathcal{L}_\lambda \le \log\big(1 + R\exp(-\mu_\lambda/\tau)\big)$. This bound is below $\log 2$ when $\mu_\lambda > \tau \log R$, i.e., $\lambda > (\rho + \tau\log R)/(\rho+\beta)$.*

A low loss thus demands more weight than ranking alone, because the loss aggregates all R negatives.

From attention to repair. Memory overcomes a frozen-query deficit ρ in four steps:

- A large logit gap directs attention to useful context (Lemmas B.2 and B.5).
- Discriminative context yields discriminative memory, $\beta > 0$ (Lemma 6.4).
- Memory repairs the ranking when its weighted margin exceeds the deficit (Thm 6.5).
- A sufficiently large λ drives InfoNCE below $\log 2$, ensuring correct ranking (Corollary 6.6).

The linear analysis rests on $\widetilde{\mathbf{z}}_q$ being an exact interpolation of $\mathbf{z}_q$ and $\overline{\mathbf{v}}$. In a Transformer, self-attention lets the inserted tokens alter every representation, so such a decomposition does not exist. We instead switch memory on gradually in input space. Let $\widetilde{\mathbf{E}}_{q,0}$ be the augmented input with padding in the memory and boundary slots, chosen so that $f_{\theta_q}(\widetilde{\mathbf{E}}_{q,0}) = \mathbf{z}_q$, and let $\mathbf{\Delta}_q$ replace the padding with $\mathbf{B}_{\mathrm{L}}, \mathbf{m}_c, \mathbf{B}_{\mathrm{R}}$. Along $\widetilde{\mathbf{E}}_q(\lambda) = \widetilde{\mathbf{E}}_{q,0} + \lambda \mathbf{\Delta}_q$, $\lambda \in [0,1]$,[1] the margins $\gamma_j(\lambda) = f_{\theta_q}(\widetilde{\mathbf{E}}_q(\lambda))^\top(\mathbf{z}^+ - \mathbf{z}_j^-)$ start at $\gamma_j(0) = \gamma_j \geq -\rho$ and have initial slopes $\gamma_j'(0) = \langle Df_{\theta_q}(\widetilde{\mathbf{E}}_{q,0})[\mathbf{\Delta}_q], \mathbf{z}^+ - \mathbf{z}_j^- \rangle$, where $Df_{\theta_q}(\mathbf{E})[\mathbf{\Delta}]$ is the directional derivative of the encoder. The worst slope $\beta = \min_j \gamma_j'(0)$ plays the role of the discriminative margin.

Theorem 6.7 (Frozen-query repair through a Transformer)**.** *If each map $\mathbf{E} \mapsto f_{\theta_q}(\mathbf{E})^\top(\mathbf{z}^+ - \mathbf{z}_j^-)$ has an H-Lipschitz gradient, then $\gamma_j(\lambda) \geq -\rho + \lambda\beta - \frac{1}{2}H\lambda^2\|\mathbf{\Delta}_q\|_F^2$ for all j and $\lambda \in [0,1]$. Hence the augmented query ranks d^+ above every negative if $\beta > \rho + \frac{1}{2}H\|\mathbf{\Delta}_q\|_F^2$.*

As in Theorem 6.5, memory repairs the frozen query when its gain β beats the deficit ρ, now plus a curvature penalty for the encoder's nonlinearity. InfoNCE directly targets such margins, and a loss below $\log 2$ certifies the ranking for any encoder (Appendix B.4). The condition is sufficient but not necessary, since H is rarely computable for a Transformer.

7 Experiments

We evaluate on eight retrieval datasets with five encoder backbones, guided by the following research questions:

RQ1 How does REMEDI compare with fine-tuning the query tower?
RQ2 How does REMEDI compare with query augmentation methods?
RQ3 How much does fine-tuning degrade out-of-domain retrieval?
RQ4 How sensitive is REMEDI to the quality of its memory?
RQ5 Does REMEDI generalize to concepts unseen in training?
RQ6 How do design choices for the adapter A_ψ affect REMEDI?

Metrics. Following prior work (Thakur et al., 2021; Karpukhin et al., 2020), we assess the retrieval accuracy for all methods using nDCG@10 as our test metric.

Datasets. We evaluate on eight datasets (Table 6 in Appendix D.1): MedQuAD (Ben Abacha & Demner-Fushman, 2019), NQ (Kwiatkowski et al., 2019), TriviaQA (Joshi et al., 2017), EntityQuestions (Sciavolino et al., 2021), HotpotQA (Yang et al., 2018), SQuAD (Rajpurkar et al., 2016), PopQA (Mallen et al., 2023), and FEVER (Thorne et al., 2018). NQ, TriviaQA, and EntityQuestions share the DPR Wikipedia corpus, where a passage is relevant if it contains a gold answer; the rest use their own corpora. MedQuAD has no standard split, so we create a random one. We defer a detailed description of datasets to Appendix D.1.

Memory construction. We use an LLM to extract concepts from the training queries (Appendix D.2). For each concept, we store five LLM-written contexts, each covering a different aspect and generated from the concept alone, followed by its top-five BM25 passages, resulting in a total of ten context documents per concept. For MedQuAD, we use the 4,741 `focus_area` labels as concepts and their 16,344 deduplicated answers as memory. The frozen document encoder encodes every context once and caches it, so the index stays unchanged.

Encoders. We use five bi-encoders (Table 7, Appendix D.1): RoBERTa-base (Liu et al., 2019; Reimers & Gurevych, 2019), E5-base-unsupervised and E5-base (Wang et al., 2022b), Nomic Embed v1 (Nussbaum et al., 2024), and Qwen3-Embedding-0.6B (Zhang et al., 2025). They span NLI-tuned, weakly supervised, and retrieval-supervised training, and both bidirectional and decoder-only architectures. Each keeps its native prefixes and pooling.

[1] Unlike the linear weight $\lambda = C/(A+C)$, this λ scales the insertion in input space.

Table 1: RQ2: test nDCG@10 (%) on four source datasets and the mean over five out-of-domain BEIR tasks (SciFact, NFCorpus, FiQA-2018, ArguAna, and TREC-COVID).

	Qwen						RoBERTa						Nomic						Overall	
Method	MQ	NQ	EQ	TQA	Mean	BEIR	MQ	NQ	EQ	TQA	Mean	BEIR	MQ	NQ	EQ	TQA	Mean	BEIR	Mean	BEIR
Frozen	70.59	29.02	29.07	38.89	41.89	61.82	27.68	6.60	4.00	8.28	11.64	18.06	74.23	35.25	28.74	42.55	45.19	54.59	32.91	44.82
LexicalPRF	50.47	22.95	29.75	40.26	35.86	49.26	36.20	**15.58**	**21.70**	**28.10**	25.39	**34.20**	51.98	23.75	30.35	41.32	36.85	46.16	32.70	43.21
VectorPRF	60.59	30.37	**30.15**	41.55	40.67	56.08	22.34	6.62	3.92	8.27	10.29	15.60	63.91	**36.28**	**30.58**	45.76	44.13	50.42	31.70	40.70
HyDE	67.90	**34.21**	28.61	**48.73**	**44.86**	58.33	26.27	14.90	10.30	24.83	19.08	21.88	61.15	33.69	25.61	**49.73**	42.54	50.34	35.49	43.51
KPR-BERT + LoRA	82.83	21.49	24.55	30.34	39.80	59.63	63.25	11.18	10.88	13.36	24.67	19.95	**82.11**	31.18	27.13	35.72	44.04	48.94	36.17	42.84
MoMA + LoRA	76.69	21.80	20.59	27.97	36.76	40.41	43.06	9.98	7.75	11.55	18.08	14.16	75.79	29.80	24.45	37.70	41.94	37.73	32.26	30.77
REMEDI	80.57	25.96	26.64	36.46	42.41	**61.51**	79.62	11.34	6.07	12.51	27.39	17.42	80.81	32.44	27.42	40.51	**45.30**	**54.19**	**38.36**	**44.37**
REMEDI + LoRA	**83.34**	21.57	23.88	31.19	40.00	59.54	**81.93**	13.70	8.90	15.47	**30.00**	14.67	81.37	31.15	26.53	36.16	43.80	51.14	37.93	41.78

Training A_ψ. REMEDI trains solely the memory adapter A_ψ using InfoNCE ($\tau = 0.05$, 64 queries per batch) and AdamW (learning rate 10^{-4}, weight decay 0.01) with 10% warm-up, cosine decay, and gradient clipping at 1.0. We train for up to 20 epochs and stop early (patience 3) based on validation retrieval. LoRA and full fine-tuning follow the same setup but use learning rates of 4×10^{-5} and 10^{-5}, respectively.

Baselines. All baselines keep the document encoder and index frozen (Appendix C). We compare against three groups. *Query-tower fine-tuning* adapts the query encoder and uses no memory: full fine-tuning (Full FT) and LoRA (Hu et al., 2022) with ranks 16 and 32. *Training-free query expansion* enriches the query without learning: LexicalPRF averages the dense vectors of the top-5 BM25 documents into the query, and HyDE (Gao et al., 2023) retrieves with the embedding of an LLM-generated hypothetical document. VectorPRF (Li et al., 2023) averages the dense vectors of the frozen query's own top-5 documents into the query. *Learned memory interfaces* read the same concept dictionary as REMEDI: KPR-BERT (Yamada et al., 2025) fuses masked-LM concept vectors into the query embedding, and MoMA (Ge et al., 2023) reads the query and memory through an added two-layer decoder; both are trained jointly with query-side LoRA-r16. We report REMEDI both alone and with the same LoRA.

7.1 RQ1: REMEDI versus Query-Tower Fine-Tuning

Table 2 compares REMEDI with query-tower fine-tuning:
(1) REMEDI wins overall without updating a single encoder weight. It posts the best average on every encoder (65.9 across encoders, vs. 63.5 for Full FT and 63.3 for LoRA-r32) and leads all adapted models in 31 of 40 cells.
(2) The weaker the encoder, the larger the gain. RoBERTa, tuned for sentence similarity, holds little query-relevant knowledge; REMEDI supplies it from memory and lifts the average from 32.7 (Full FT) to 41.1.
(3) A strong encoder still benefits. Qwen3 knows far more (70.6 frozen on MedQuAD vs. 27.7 for RoBERTa), yet REMEDI still wins six of eight datasets.
(4) Memory helps most on entity-centric queries. REMEDI beats every adapted model on NQ, EntityQuestions, and HotpotQA for all encoders, although frozen E5-base and Nomic remain ahead on NQ and HotpotQA. MedQuAD is the exception: on the four retrieval-trained encoders, fine-tuning leads by 0.4 to 4.0 points.

Table 2: RQ1: comparing nDCG@10 for REMEDI vs. fine-tuning. Best is in **bold**, and the second best is underlined. MQ: MedQuAD, EQ: EntityQuestions, TQA: TriviaQA, HQA: HotpotQA.

	Method	MQ	NQ	EQ	TQA	HQA	SQuAD	PopQA	FEVER	Avg.
RoBERTa	Frozen	27.68	15.6	8.7	16.6	25.2	31.9	8.1	38.3	21.5
	LoRA-r16	45.22	20.9	16.5	19.8	28.2	39.1	14.6	56.1	30.1
	LoRA-r32	46.57	21.5	16.6	20.0	28.3	39.5	14.9	56.3	30.5
	Full FT	54.04	23.3	18.7	21.8	29.9	**40.8**	16.0	56.9	32.7
	REMEDI	**79.62**	**25.3**	**29.6**	**27.0**	**36.9**	37.2	**23.3**	**70.0**	**41.1**
E5-unsup.	Frozen	69.97	48.4	60.2	45.0	65.9	60.9	44.3	91.6	60.8
	LoRA-r16	79.38	48.6	73.3	**53.0**	75.4	**65.6**	57.3	95.1	68.5
	LoRA-r32	80.40	48.1	73.6	52.8	75.4	65.3	56.8	**95.2**	68.5
	Full FT	**80.86**	47.7	73.7	52.7	75.7	65.2	56.1	95.1	68.4
	REMEDI	80.48	**48.9**	**74.7**	52.4	**76.6**	65.2	**57.7**	95.1	**68.9**
E5-base	Frozen	69.41	65.8	66.5	56.1	75.6	69.3	54.9	92.6	68.8
	LoRA-r16	77.68	61.0	74.3	57.7	74.4	68.9	62.3	94.8	71.4
	LoRA-r32	78.18	60.3	74.3	57.1	74.3	68.2	62.4	95.0	71.2
	Full FT	**78.86**	58.9	73.5	56.2	73.1	67.5	61.9	94.7	70.6
	REMEDI	74.89	**63.7**	**75.9**	**58.9**	**75.5**	**69.8**	**64.3**	**95.1**	**72.3**
Nomic	Frozen	74.23	65.6	70.7	59.2	82.1	76.7	61.5	92.5	72.8
	LoRA-r16	81.26	62.1	77.5	60.9	81.1	74.1	64.9	95.3	74.6
	LoRA-r32	**81.30**	62.0	77.8	**61.1**	80.8	74.0	64.9	95.4	74.7
	Full FT	81.26	61.5	77.6	60.7	80.5	73.8	65.7	95.3	74.5
	REMEDI	80.81	**63.1**	**78.3**	**61.1**	**81.7**	**74.9**	**66.7**	**95.6**	**75.3**
Qwen3	Frozen	70.59	56.5	68.2	56.7	76.5	74.6	54.2	94.2	68.9
	LoRA-r16	83.22	55.5	73.3	58.7	75.8	74.3	58.6	94.7	71.8
	LoRA-r32	**83.32**	55.6	73.1	58.6	75.5	74.2	58.9	95.0	71.8
	Full FT	82.71	55.0	73.1	58.2	75.9	72.8	**59.5**	95.0	71.5
	REMEDI	80.57	**56.8**	**73.8**	**59.0**	**77.1**	**74.9**	59.2	**95.5**	**72.1**

Table 4: RQ4: nDCG@10 under noisy memory. (a) A Qwen3 adapter trained on clean memory is tested with k of ten slots replaced by other concepts' random contexts, or with only its five LLM or BM25 slots; change from Clean. (b) Each column trains and tests an NQ adapter on memory with L/B/R = numbers of LLM, BM25, and random slots (e.g., 4/4/2 is 20% noise). ± indicates the change from Frozen.

(a) Test-time noise, clean-trained Qwen3								
	Frozen	Clean	k=2	k=4	k=8	k=10	LLM	BM25
MQ	70.59	80.40	+0.00	−0.04	−0.94	−1.38	–	–
NQ	29.03	30.76	−0.21	−0.79	−3.59	−11.22	−0.31	−1.28

(b) Noisy training on NQ, L/B/R slots								
	Frozen	5/0/0	0/5/0	5/5/0	4/4/2	3/2/5	1/1/8	0/0/10
RoBERTa	15.56	+14.43	+20.13	**+21.82**	+20.63	+19.53	+15.63	+3.02
Qwen3	60.04	−0.34	+0.24	**+1.20**	+0.27	−0.62	−0.12	−0.67

7.2 RQ2: Comparison with Query Augmentation

RQ2 compares REMEDI with methods that enrich the query while keeping the index fixed (Table 1):
(1) REMEDI alone gains in domain at almost no out-of-domain cost. It posts the best overall mean (38.36 vs. 32.91 frozen) and the best BEIR mean among query augmentation methods (44.37, within 0.45 of frozen).
(2) Specialized domains need learned memory. On MedQuAD, every learned module beats all three training-free methods on every encoder, and REMEDI lifts RoBERTa from 27.68 to 79.62. Training-free expansion even hurts: on Qwen and Nomic, all three methods fall below frozen.
(3) Open-domain QA favors LLM knowledge. HyDE gains most on TriviaQA (+9.8 on Qwen, +7.2 on Nomic), but needs an LLM call per query and loses on MedQuAD and BEIR. On these strong encoders, every learned module trails frozen on NQ, EntityQuestions, and TriviaQA; REMEDI trails least.
(4) With a weak encoder, lexical methods win on entity-centric queries. For RoBERTa, LexicalPRF beats every method on NQ, EntityQuestions, and TriviaQA, lifting EntityQuestions from 4.00 to 21.70, since these queries hinge on entity names that BM25 matches exactly.
(5) Tuning the query tower trades generality for in-domain gains. Adding LoRA raises REMEDI's MedQuAD score but lowers its BEIR score on every encoder, and MoMA drops overall BEIR from 44.82 to 30.77, as RQ3 examines.

7.3 RQ3: BEIR Regression

Tuning the query tower buys in-domain gains at an out-of-domain cost, which REMEDI avoids (Table 3). On Qwen and Nomic, LoRA and Full FT raise MedQuAD by 7.0 to 12.7 points but lower BEIR by 2.2 to 5.2. REMEDI raises MedQuAD by 6.6 to 10.0 points on these encoders, yet stays within 0.64 points of frozen on BEIR, because most BEIR queries match no concept and take the frozen path. Adding LoRA to REMEDI brings the loss back (2.3 to 3.5 points), so tuning the encoder, not injecting memory, causes the regression. Only the weak RoBERTa gains on BEIR from tuning (+1.5 to +1.7).

Table 3: RQ3: change from Frozen in nDCG@10 on MedQuAD (MQ) and BEIR.

	Qwen		**RoBERTa**		**Nomic**	
Method	MQ	BEIR	MQ	BEIR	MQ	BEIR
Frozen (abs.)	70.59	61.82	27.68	18.06	74.23	54.59
LoRA-r16	+12.63	−3.85	+17.54	+1.49	+7.03	−5.00
LoRA-r32	+12.73	−2.61	+18.89	**+1.72**	+7.07	−5.22
Full FT	+12.12	−2.23	+26.36	+1.65	+7.03	−2.89
REMEDI	+9.98	**−0.23**	+51.94	−0.64	+6.58	**−0.40**
REMEDI + LoRA	**+12.75**	−2.28	**+54.25**	−3.39	**+7.14**	−3.45

7.4 RQ4: Sensitivity to Memory Quality

We test two settings. **(a)** *Test-time noise:* an adapter trained on clean memory sees k of each concept's ten slots replaced by other concepts' contexts. **(b)** *Noisy training:* the adapter is trained and evaluated with L LLM, B BM25, and R random slots (L/B/R; Table 4).
(1) REMEDI tolerates heavy test-time corruption. With four of ten slots replaced by noise, it loses only 0.04 points on MedQuAD and 0.79 on NQ, and falls below frozen on NQ only when noise fills most slots ($k \geq 8$). Even with fully corrupted memory, it stays 8.4 points above frozen on MedQuAD: the substitutes are still health answers, so the adapter extracts domain signal.
(2) LLM contexts and BM25 passages complement each other. At test time, removing BM25 costs 0.31 points on NQ, and removing LLM contexts costs 1.28. In training, BM25 beats LLM contexts on both encoders, LLM contexts alone push Qwen3 below frozen, and the combination wins.

Table 5: RQ5: nDCG@10 on MedQuAD concepts unseen in training. Help/Harm: % of queries scored above/below Frozen. Retrospective rows reuse the original checkpoints; held-out rows retrain on a split with 488 unseen test concepts. Per encoder, the largest Δ is **bold** and the second is underlined.

	Nomic				Qwen3				RoBERTa			
Evaluation set	Frozen	REMEDI	Δ	Help/Harm	Frozen	REMEDI	Δ	Help/Harm	Frozen	REMEDI	Δ	Help/Harm
Retrospective, original split												
Unseen concepts (840 q)	93.16	96.85	+3.69	8.9/1.4	93.40	96.71	+3.31	9.8/1.6	53.87	99.20	+45.33	56.2/0.0
literal-evidence-free (553 q)	95.80	98.07	+2.27	6.2/1.1	94.58	97.91	+3.33	9.0/0.9	63.40	99.67	+36.26	45.8/0.0
Concept-held-out split												
Query macro (1,640 q)	76.16	82.50	**+6.34**	20.6/4.1	71.90	82.91	**+11.00**	31.2/5.5	30.84	84.61	**+53.77**	70.0/2.3
Concept macro (488 c)	85.43	90.45	+5.02	–	82.70	90.29	+7.59	–	42.11	92.92	+50.81	–

(3) The memory, not the extra parameters, drives the gain. Under noisy training, REMEDI degrades gracefully: RoBERTa retains $+19.5$ of its $+21.8$ points at 50% noise and $+15.6$ at 80%, yet collapses to $+3.0$ once random passages fill every slot. Qwen3's small NQ gains vanish beyond 20% noise.

7.5 RQ5: Generalization to Unseen Concepts

Since users add concepts after training, REMEDI must also read memory for concepts it never saw. We test this on MedQuAD in two ways (Table 5). *Retrospectively*, we evaluate the original checkpoints on the 840 test queries whose concept is absent from training, and on a stricter subset of 553 *literal-evidence-free* queries that share neither their answer nor any concept mention with training. *Prospectively*, we retrain on a split that holds out 488 entire concepts (1,640 test queries).

In both settings, REMEDI improves over the frozen baseline for every encoder. These gains are not due to leakage, as they persist on the literal-evidence-free subset. Nor are they driven by a few frequent concepts, since per-concept and per-query averages agree. Memory also rarely hurts: far more queries improve than degrade. Hence, these results show that the adapter learns *how* to read memory rather than memorizing *what* each concept's memory says.

7.6 RQ6: The Design Space for Adapter A_ψ

We ablate three design choices of the adapter A_ψ, varying one at a time. *(i) Attention over contexts:* by default, A_ψ attends over a concept's contexts once, using the pooled query embedding; token-conditioned attention instead lets each query token attend separately and then mean-pools the outputs. The pooled default is both cheaper and more accurate, by up to 3.01 points on RoBERTa. *(ii) Boundary tokens:* learned embeddings that delimit the inserted memory token. Removing them costs Qwen3 0.72 points and leaves RoBERTa unaffected. *(iii) Memory budget:* the number of contexts stored per concept. Qwen3 saturates at three contexts, whereas RoBERTa loses up to 12.97 points with fewer. We defer a detailed discussion of these choices and the accompanying results to Appendix E.1.

7.7 Attention Analysis

To understand *how* REMEDI uses its memory, we perform a mechanistic analysis of the adapter's attention on the NQ adapters of Table 4 (8,777 activated query–concept pairs, ten slots each). The adapter reads selectively: at 80% noise, the two clean slots receive 93% of Qwen3's attention and 77% of RoBERTa's, versus 20% under uniform weighting (details in Appendix E.3).

8 Conclusion

A dense retriever does not need to memorize a concept to retrieve for it. REMEDI stores knowledge in a user-editable concept dictionary and trains a small adapter to read from it, leaving the encoder and index untouched. Our theory shows when memory provably lifts a query above its hard negatives. Across five encoders and eight datasets, REMEDI beats full fine-tuning and LoRA on every encoder, stays within 0.64 points of the frozen model on BEIR, generalizes to unseen concepts, and tolerates noisy memory. External memory is thus a practical alternative to fine-tuning when a deployed retriever must adapt without losing its general-purpose behavior.

AI USE STATEMENT

We used large language models in three parts of our pipeline, all described in the paper. (i) *Data construction:* GPT-4o extracts concept spans from training queries and writes the five query-independent synopses stored in each concept's memory (Appendix D.2); the generator never sees the query, answer, or retrieved documents. (ii) *Baselines:* HyDE uses LLMs to generate hypothetical documents. (iii) *Qualitative analysis:* GPT-4o labels retrieved passages for the case studies (Section 2, Appendix E.3). We also used AI assistants to edit the paper for readability, and make the prose concise. We did not use generative AI to write or debug code, or to formulate the method or the theoretical results. All authors reviewed all AI-assisted content and take full responsibility for it.

ETHICS STATEMENT

All datasets are public benchmarks, and our experiments involve no human subjects. MedQuAD contains consumer-health questions, and part of our concept memory is generated by an LLM, which can produce inaccurate medical statements. REMEDI is a retrieval component, not a source of medical advice. Because its memory is user-editable, erroneous entries can be inspected and corrected without retraining, but deployments in sensitive domains should curate memory with domain experts.

REPRODUCIBILITY STATEMENT

Appendix D gives datasets, splits, encoders, optimization settings, and evaluation details; Appendix D.2 gives the exact synopsis-generation prompts; proofs of all theoretical results are in Appendix B. We will release cleaned-up code, concept dictionaries, and cached memory upon publication.

REFERENCES

Asma Ben Abacha and Dina Demner-Fushman. A question-entailment approach to question answering. *BMC Bioinformatics*, 20(1):511, 2019. doi: 10.1186/s12859-019-3119-4.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pp. 2206–2240, 2022.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. Autoregressive entity retrieval. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://openreview.net/forum?id=5k8F6UU39V.

Thibault Févry, Livio Baldini Soares, Nicholas FitzGerald, Eunsol Choi, and Tom Kwiatkowski. Entities as experts: Sparse memory access with entity supervision. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4937–4951, 2020. doi: 10.18653/v1/2020.emnlp-main.400.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1762–1777, 2023. doi: 10.18653/v1/2023.acl-long.99.

Suyu Ge, Chenyan Xiong, Corby Rosset, Arnold Overwijk, Jiawei Han, and Paul Bennett. Augmenting zero-shot dense retrievers with plug-in mixture-of-memories. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1796–1812, 2023. doi: 10.18653/v1/2023.emnlp-main.111.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Realm: retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pp. 2790–2799, 2019.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=jKN1pXi7b0.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251):1–43, 2023. URL https://jmlr.org/papers/v24/23-0037.html.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 1601–1611, 2017. doi: 10.18653/v1/P17-1147.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. Large language models struggle to learn long-tail knowledge. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pp. 15696–15707, 2023.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, 2020. doi: 10.18653/v1/2020.emnlp-main.550.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural Questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026/.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 9459–9474, 2020.

Hang Li, Ahmed Mourad, Shengyao Zhuang, Bevan Koopman, and Guido Zuccon. Pseudo relevance feedback with deep language models and dense retrievers: Successes and pitfalls. *ACM Transactions on Information Systems*, 41(3):1–40, 2023. doi: 10.1145/3570724.

Hang Li, Xiao Wang, Bevan Koopman, and Guido Zuccon. Pseudo relevance feedback is enough to close the gap between small and large dense retrieval models. *arXiv preprint arXiv:2503.14887*, 2025. URL https://arxiv.org/abs/2503.14887.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. URL https://arxiv.org/abs/1907.11692.

Iain Mackie, Shubham Chatterjee, and Jeffrey Dalton. Generative relevance feedback with large language models. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2026–2031, 2023. doi: 10.1145/3539618.3591992.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 9802–9822, 2023. doi: 10.18653/v1/2023.acl-long.546.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernández Ábrego, Ji Ma, Vincent Y. Zhao, Yi Luan, Keith B. Hall, Ming-Wei Chang, and Yinfei Yang. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9844–9855, 2022. doi: 10.18653/v1/2022.emnlp-main.669.

Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. Document expansion by query prediction. *arXiv preprint arXiv:1904.08375*, 2019. URL https://arxiv.org/abs/1904.08375.

Zach Nussbaum, John X Morris, Brandon Duderstadt, and Andriy Mulyar. Nomic embed: Training a reproducible long context text embedder. *arXiv preprint arXiv:2402.01613*, 2024.

Oded Ovadia, Menachem Brief, Moshik Mishaeli, and Oren Elisha. Fine-tuning or retrieval? comparing knowledge injection in LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 237–250, 2024. doi: 10.18653/v1/2024.emnlp-main.15.

Matthew E. Peters, Mark Neumann, Robert L. Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. Knowledge enhanced contextual word representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP-IJCNLP)*, pp. 43–54, 2019. doi: 10.18653/v1/D19-1005.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP-IJCNLP)*, pp. 2463–2473, 2019. doi: 10.18653/v1/D19-1250.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 2383–2392, 2016.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, 2019. doi: 10.18653/v1/D19-1410.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, April 2009. ISSN 1554-0669. doi: 10.1561/1500000019. URL https://doi.org/10.1561/1500000019.

Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. Simple entity-centric questions challenge dense retrievers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6138–6148, 2021. doi: 10.18653/v1/2021.emnlp-main.496.

Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. End-to-end memory networks. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, pp. 2440–2448, Cambridge, MA, USA, 2015. MIT Press.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the 35th Conference on Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2021. URL https://openreview.net/forum?id=wCu6T5xFjeJ.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. Fever: a large-scale dataset for fact extraction and verification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 809–819, 2018.

Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. GPL: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 2345–2360, 2022a. doi: 10.18653/v1/2022.naacl-main.168.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022b. URL https://arxiv.org/abs/2212.03533.

Liang Wang, Nan Yang, and Furu Wei. Query2doc: Query expansion with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9414–9423, 2023. doi: 10.18653/v1/2023.emnlp-main.585.

Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. In *International Conference on Learning Representations (ICLR)*, 2015. URL https://arxiv.org/abs/1410.3916.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://openreview.net/forum?id=zeFrfgyZln.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6442–6454, 2020. doi: 10.18653/v1/2020.emnlp-main.523.

Ikuya Yamada, Ryokan Ri, Takeshi Kojima, Yusuke Iwasawa, and Yutaka Matsuo. Dynamic injection of entity knowledge into dense retrievers. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, 2025. doi: 10.18653/v1/2025.findings-emnlp.915.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2369–2380, 2018. doi: 10.18653/v1/D18-1259.

HongChien Yu, Chenyan Xiong, and Jamie Callan. Improving query representations for dense retrieval with pseudo relevance feedback. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 3592–3596, 2021. doi: 10.1145/3459637.3482124.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025. URL https://arxiv.org/abs/2506.05176.

Zhengyan Zhang, Zhiyuan Zeng, Yankai Lin, Huadong Wang, Deming Ye, Chaojun Xiao, Xu Han, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. Plug-and-play knowledge injection for pre-trained language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 10641–10658, 2023. doi: 10.18653/v1/2023.acl-long.594.

Shengyao Zhuang, Xueguang Ma, Bevan Koopman, Jimmy Lin, and Guido Zuccon. PromptReps: Prompting large language models to generate dense and sparse representations for zero-shot document retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4375–4391, 2024. doi: 10.18653/v1/2024.emnlp-main.250.

A Extended Related Work

Dense retrieval and parametric adaptation. Dual encoders embed each query and document into a single vector and rank candidates by similarity (Karpukhin et al., 2020; Reimers & Gurevych, 2019; Xiong et al., 2021; Izacard et al., 2022; Ni et al., 2022). Frontier language models represent frequent entities far more reliably than tail entities (Petroni et al., 2019; Kandpal et al., 2023; Mallen et al., 2023); dense retrievers likewise generalize poorly to unseen entities (Sciavolino et al., 2021), consistent with the domain-shift sensitivity documented in BEIR (Thakur et al., 2021). Existing work closes this gap either by updating the encoder or by giving the query external context. Full fine-tuning, adapters, and LoRA (Houlsby et al., 2019; Hu et al., 2022) take the first route; even parameter-efficient variants encode knowledge through weight updates and risk shifting retrieval behavior outside the target domain. REMEDI instead leaves the encoder untouched and supplies external context to the query.

Query enrichment and feedback. A complementary line of work enriches queries directly rather than adapting the encoder. Pseudo-relevance feedback (PRF) retrieves documents and uses them to refine the query (Li et al., 2023; Yu et al., 2021), while generative expansion prompts a language model to produce additional query text (Nogueira et al., 2019; Gao et al., 2023; Wang et al., 2023; Mackie et al., 2023; Zhuang et al., 2024). Both typically rewrite the query as a whole or synthesize a query-level representation, adding an online retrieval pass or LLM call. PRF further assumes that its first-pass ranking is informative: when an unfamiliar concept causes the initial search to miss relevant documents, feedback cannot supply the missing knowledge. ANCE-PRF (Yu et al., 2021) and PromptPRF (Li et al., 2025) keep the document index fixed but still update the query tower parameters.

Non-parametric memory. A separate line of work equips models with non-parametric memory that stores knowledge outside the encoder's parameters (Weston et al., 2015; Sukhbaatar et al., 2015; Guu et al., 2020; Lewis et al., 2020; Borgeaud et al., 2022; Izacard et al., 2023). Most such systems use memory to improve what a *generator* produces; REMEDI instead improves the *retriever's* query representation. The closest approach, MoMA (Ge et al., 2023), augments the query with documents retrieved from mixed corpora, but requires jointly training an augmentation retriever and an encoder–decoder backbone, maintaining a second index, and re-encoding the corpus. REMEDI avoids this machinery altogether.

Entity-aware representations and knowledge injection. Because REMEDI indexes memory by concept, it also relates to entity-aware representation learning, which either builds in entity-specific parameters or supervision (Peters et al., 2019; Yamada et al., 2020; Févry et al., 2020; De Cao et al., 2021), or maps external knowledge into a frozen backbone (Zhang et al., 2023). The closest method in this stream, KPR (Yamada et al., 2025), augments a frozen retriever with learned entity embeddings and therefore cannot represent an unseen concept without retraining. REMEDI, on the other hand, represents each concept with user-editable documents, allowing a user to add an unseen concept at inference time.

B Deferred Proofs

This appendix collects the proofs of the results stated in Section 6. Main-text statements are restated here with their original numbers, and four supporting lemmas on attention mass, logit gaps, and a sufficient condition for discrimination are stated here for the first time; we give all arguments in full. The proofs are grouped to mirror the structure of the analysis: the augmented query, improvement in positive similarity, discriminative memory contexts, the InfoNCE guarantees, and the extension to Transformers.

B.1 The Augmented Query in the Linear Model

Lemma 6.1 (Augmented query as an interpolation)**.** *If $C > 0$, then $\|\overline{\mathbf{v}}\|_2 \le 1$, $\lambda \in (0, 1)$, and*

$$\widetilde{\mathbf{z}}_q = \frac{(1-\lambda)\mathbf{z}_q + \lambda\overline{\mathbf{v}}}{\|(1-\lambda)\mathbf{z}_q + \lambda\overline{\mathbf{v}}\|_2} \tag{3}$$

whenever the numerator is nonzero. If $C = 0$, then $\widetilde{\mathbf{z}}_q = \mathbf{z}_q$.

Proof of Lemma 6.1. The augmented input contains the original query-token embeddings together with $\mathbf{B}_{\mathrm{L}}$, $\mathbf{m}_c$, and $\mathbf{B}_{\mathrm{R}}$. Under the linear encoder model, its unnormalized representation is

$$\widetilde{\mathbf{u}}_q = W_f \left(\sum_{t=1}^{L} \mathbf{e}_t + \mathbf{B}_{\mathrm{L}} + \mathbf{m}_c + \mathbf{B}_{\mathrm{R}} \right) \tag{5}$$

$$= \mathbf{u}_q + W_f \mathbf{B}_{\mathrm{L}} + W_f \mathbf{m}_c + W_f \mathbf{B}_{\mathrm{R}}. \tag{6}$$

Substituting the definition of $\mathbf{m}_c$ gives

$$W_f \mathbf{m}_c = \sum_{k=1}^{K} \alpha_k W_f W_O W_V \mathbf{h}_k. \tag{7}$$

Since $\sum_{k=1}^{K} \alpha_k = 1$, the boundary contribution can be written as

$$W_f \mathbf{B}_{\mathrm{L}} + W_f \mathbf{B}_{\mathrm{R}} = \sum_{k=1}^{K} \alpha_k \left(W_f \mathbf{B}_{\mathrm{L}} + W_f \mathbf{B}_{\mathrm{R}} \right). \tag{8}$$

Combining these expressions yields

$$\widetilde{\mathbf{u}}_q = \mathbf{u}_q + \sum_{k=1}^{K} \alpha_k \mathbf{v}_k. \tag{9}$$

For $C > 0$, each normalized effective contribution satisfies $\|\widehat{\mathbf{v}}_k\|_2 \leq 1$. Because $\overline{\mathbf{v}}$ is their convex combination, $\|\overline{\mathbf{v}}\|_2 \leq 1$. Using $\mathbf{u}_q = A\mathbf{z}_q$ and $\sum_k \alpha_k \mathbf{v}_k = C\overline{\mathbf{v}}$, we obtain

$$\widetilde{\mathbf{u}}_q = A\mathbf{z}_q + C\overline{\mathbf{v}} \tag{10}$$

$$= (A + C) \left[\frac{A}{A + C} \mathbf{z}_q + \frac{C}{A + C} \overline{\mathbf{v}} \right]. \tag{11}$$

The positive scalar $A + C$ cancels under ℓ_2 normalization, which gives Eq. 3. If $C = 0$, then $\mathbf{v}_k = \mathbf{0}$ for every k, and Eq. 9 gives $\widetilde{\mathbf{u}}_q = \mathbf{u}_q$. Hence, $\widetilde{\mathbf{z}}_q = \mathbf{z}_q$. □

B.2 Improvement in Positive Similarity

Lemma 6.2. *If $C > 0$ and $0 \leq a < b$, then the augmented query strictly improves positive similarity: $\widetilde{\mathbf{z}}_q^\top \mathbf{z}^+ > \mathbf{z}_q^\top \mathbf{z}^+$.*

Proof of Lemma 6.2. Let $\mathbf{y} = (1 - \lambda)\mathbf{z}_q + \lambda\overline{\mathbf{v}}$ denote the unnormalized interpolation in Eq. 3. Since $\|\mathbf{z}_q\|_2 = 1$ and $\|\overline{\mathbf{v}}\|_2 \leq 1$,

$$\|\mathbf{y}\|_2 \leq (1 - \lambda)\|\mathbf{z}_q\|_2 + \lambda\|\overline{\mathbf{v}}\|_2 \leq 1. \tag{12}$$

Moreover,

$$\mathbf{y}^\top \mathbf{z}^+ = (1 - \lambda)\mathbf{z}_q^\top \mathbf{z}^+ + \lambda\overline{\mathbf{v}}^\top \mathbf{z}^+ \tag{13}$$

$$= (1 - \lambda)a + \lambda b = a + \lambda(b - a) > a \geq 0. \tag{14}$$

Thus, $\mathbf{y} \neq \mathbf{0}$. Dividing its positive inner product with $\mathbf{z}^+$ by a norm no larger than one gives

$$\widetilde{\mathbf{z}}_q^\top \mathbf{z}^+ = \frac{\mathbf{y}^\top \mathbf{z}^+}{\|\mathbf{y}\|_2} \geq \mathbf{y}^\top \mathbf{z}^+ > a = \mathbf{z}_q^\top \mathbf{z}^+. \tag{15}$$

□

Let $s_q = \mathbf{z}_q^\top \mathbf{z}^+$ be the frozen query's positive similarity. Suppose a useful context with index $\star$ scores higher, $s_\star = \widehat{\mathbf{v}}_\star^\top \mathbf{z}^+ > s_q \geq 0$, while every other context scores $s_k = \widehat{\mathbf{v}}_k^\top \mathbf{z}^+ \geq s_{\min}$ for some $s_{\min} < s_q$. The next lemma bounds the attention that $\star$ needs to raise the query-positive similarity.

Lemma B.1. *If $\alpha_\star > (s_q - s_{\min})/(s_\star - s_{\min})$, then $\overline{\mathbf{v}}^\top \mathbf{z}^+ > s_q$ holds.*

The threshold on $\alpha_\star$ rises with misleading distractors (large $s_q - s_{\min}$) and falls with a strong useful context (large $s_\star - s_{\min}$).

Proof of Lemma B.1. The positive similarity of the attended memory direction satisfies

$$\overline{\mathbf{v}}^\top \mathbf{z}^+ = \alpha_\star \widehat{\mathbf{v}}_\star^\top \mathbf{z}^+ + \sum_{k \neq \star} \alpha_k \widehat{\mathbf{v}}_k^\top \mathbf{z}^+ \tag{16}$$

$$\geq \alpha_\star s_\star + (1 - \alpha_\star) s_{\min}. \tag{17}$$

The right-hand side exceeds s_q exactly when

$$\alpha_\star s_\star + (1 - \alpha_\star) s_{\min} > s_q \tag{18}$$

$$\iff \quad \alpha_\star (s_\star - s_{\min}) > s_q - s_{\min} \tag{19}$$

$$\iff \quad \alpha_\star > \frac{s_q - s_{\min}}{s_\star - s_{\min}}. \tag{20}$$

The denominator is positive because $s_\star > s_q > s_{\min}$. The stated condition therefore gives $\overline{\mathbf{v}}^\top \mathbf{z}^+ > s_q$, and Lemma 6.2 completes the proof. □

Since attention is a softmax, the condition of Lemma B.1 can also be posed as a logit gap.

Lemma B.2. *If $K > 1$ and $\ell_\star - \ell_k \geq \delta$ for all $k \neq \star$, then $\alpha_\star \geq [1 + (K-1)\exp(-\delta/\tau_{\text{att}})]^{-1}$. Hence the condition of Lemma B.1 holds whenever $\delta > \tau_{\text{att}} \log\big((K-1)(s_q - s_{\min})/(s_\star - s_q)\big)$.*

Proof of Lemma B.2. The assumed logit gap implies that every other memory context satisfies $\ell_k \leq \ell_\star - \delta$. Since the exponential function is increasing,

$$\exp(\ell_k/\tau_{\text{att}}) \leq \exp\left(\frac{\ell_\star - \delta}{\tau_{\text{att}}}\right) \qquad \text{for every } k \neq \star. \tag{21}$$

There are $K - 1$ such contexts. Their total contribution to the softmax denominator is therefore bounded by

$$\sum_{k \neq \star} \exp(\ell_k/\tau_{\text{att}}) \leq (K-1)\exp\left(\frac{\ell_\star - \delta}{\tau_{\text{att}}}\right). \tag{22}$$

The attention mass of the useful attended memory context is

$$\alpha_\star = \frac{\exp(\ell_\star/\tau_{\text{att}})}{\exp(\ell_\star/\tau_{\text{att}}) + \sum_{k \neq \star} \exp(\ell_k/\tau_{\text{att}})} \tag{23}$$

$$\geq \frac{\exp(\ell_\star/\tau_{\text{att}})}{\exp(\ell_\star/\tau_{\text{att}}) + (K-1)\exp((\ell_\star - \delta)/\tau_{\text{att}})} \tag{24}$$

$$= \frac{1}{1 + (K-1)\exp(-\delta/\tau_{\text{att}})}. \tag{25}$$

Thus, it remains to determine when this lower bound exceeds the attention threshold $\alpha_\star > (s_q - s_{\min})/(s_\star - s_{\min})$ of Lemma B.1. Substituting the two expressions gives

$$\frac{1}{1 + (K-1)\exp(-\delta/\tau_{\text{att}})} > \frac{s_q - s_{\min}}{s_\star - s_{\min}} \tag{26}$$

$$\iff \quad s_\star - s_q > (K-1)(s_q - s_{\min})\exp(-\delta/\tau_{\text{att}}) \tag{27}$$

$$\iff \quad \delta > \tau_{\text{att}} \log\left(\frac{(K-1)(s_q - s_{\min})}{s_\star - s_q}\right). \tag{28}$$

The final condition therefore guarantees sufficient attention on the useful memory context and, by Lemma B.1, improves similarity to the positive document. □

Remark B.3 (Positive similarity alone is insufficient). Higher positive similarity need not improve retrieval, since memory can raise a negative's similarity even more. With unit vectors $\mathbf{z}^+ = (1, 0)$, $\mathbf{z}_q = (0.6, -0.8)$, and $\widehat{\mathbf{v}}_\star = \mathbf{z}_1^- = (0.8, 0.6)$, placing all attention on context $\star$ (so $\overline{\mathbf{v}} = \widehat{\mathbf{v}}_\star$) and letting memory dominate the interpolation ($\lambda \to 1$) drives the augmented query toward $\widehat{\mathbf{v}}_\star$: positive similarity rises from $\mathbf{z}_q^\top \mathbf{z}^+ = 0.6$ to $\widehat{\mathbf{v}}_\star^\top \mathbf{z}^+ = 0.8$, yet the positive–negative margin falls from $\mathbf{z}_q^\top(\mathbf{z}^+ - \mathbf{z}_1^-) = 0.6$ to $\widehat{\mathbf{v}}_\star^\top(\mathbf{z}^+ - \mathbf{z}_1^-) = 0.8 - 1 = -0.2$, so the positive document drops below the negative.

B.3 Discriminative Memory Contexts

Lemma 6.3. *Define the worst-case margin of context $\star$ as $\omega_\star := \min_{1\le j\le R} \widehat{\mathbf{v}}_\star^\top(\mathbf{z}^+ - \mathbf{z}_j^-)$, and call context $\star$ discriminative if $\omega_\star > 0$. Then $\omega_\star = \|\widehat{\mathbf{v}}_\star\|_2(s_\star^+ - s_\star^-)$, so $\omega_\star > 0$ iff $s_\star^+ > s_\star^-$: the context is discriminative exactly when it scores the positive above every negative.*

Proof of Lemma 6.3. For every negative document $\mathbf{z}_j^-$,

$$\widehat{\mathbf{v}}_\star^\top(\mathbf{z}^+ - \mathbf{z}_j^-) = \|\widehat{\mathbf{v}}_\star\|_2\, \mathbf{r}_\star^\top(\mathbf{z}^+ - \mathbf{z}_j^-) \tag{29}$$

$$= \|\widehat{\mathbf{v}}_\star\|_2 \left(s_\star^+ - \mathbf{r}_\star^\top \mathbf{z}_j^-\right). \tag{30}$$

Taking the minimum over j is equivalent to subtracting the largest context–negative similarity. Therefore,

$$\omega_\star = \|\widehat{\mathbf{v}}_\star\|_2 \left(s_\star^+ - \max_{1\le j\le R} \mathbf{r}_\star^\top \mathbf{z}_j^-\right) \tag{31}$$

$$= \|\widehat{\mathbf{v}}_\star\|_2 (s_\star^+ - s_\star^-). \tag{32}$$

Because $\|\widehat{\mathbf{v}}_\star\|_2 > 0$, this margin is positive exactly when $s_\star^+ > s_\star^-$. □

While Lemma 6.3 is exact, the next lemma gives a simpler sufficient condition.

Lemma B.4. *Let $s = \mathbf{r}_\star^\top \mathbf{z}^+$ and $\kappa = \max_{1\le j\le R} (\mathbf{z}^+)^\top \mathbf{z}_j^- < 1$ (positive–hardest-negative similarity). If $s > \frac{3+\kappa}{4}$, the context is discriminative, with $\omega_\star \ge \|\widehat{\mathbf{v}}_\star\|_2 \big[(1-\kappa) - 2\sqrt{(1-\kappa)(1-s)}\big] > 0$.*

Proof of Lemma B.4. For any two unit vectors $\mathbf{x}$ and $\mathbf{y}$, $\|\mathbf{x}-\mathbf{y}\|_2^2 = 2(1-\mathbf{x}^\top\mathbf{y})$. Hence, the distance between the memory-context direction and the positive document is $\eta_+ := \|\mathbf{r}_\star - \mathbf{z}^+\|_2 = \sqrt{2(1-s)}$. Moreover, because $(\mathbf{z}^+)^\top \mathbf{z}_j^- \le \kappa$ for every j, the distance between the positive document and each negative is at least $\eta_- := \sqrt{2(1-\kappa)}$.

The condition $s > \frac{3+\kappa}{4}$ is equivalent to $1 - s < (1-\kappa)/4$, and hence to $\eta_+ < \eta_-/2$. The triangle inequality now gives, for every candidate negative,

$$\|\mathbf{r}_\star - \mathbf{z}_j^-\|_2 \ge \|\mathbf{z}^+ - \mathbf{z}_j^-\|_2 - \|\mathbf{r}_\star - \mathbf{z}^+\|_2 \ge \eta_- - \eta_+ > \eta_+. \tag{33}$$

Thus, $\mathbf{r}_\star$ is closer to the positive document than to every candidate negative. Since all of these vectors have unit norm, this is equivalent to $\mathbf{r}_\star^\top \mathbf{z}^+ > \mathbf{r}_\star^\top \mathbf{z}_j^-$ for every j. By Lemma 6.3, the memory context is therefore discriminative.

For the quantitative bound, note that

$$\mathbf{r}_\star^\top(\mathbf{z}^+ - \mathbf{z}_j^-) = \frac{1}{2}\left(\|\mathbf{r}_\star - \mathbf{z}_j^-\|_2^2 - \|\mathbf{r}_\star - \mathbf{z}^+\|_2^2\right) \tag{34}$$

$$\ge \frac{1}{2}\left[(\eta_- - \eta_+)^2 - \eta_+^2\right], \tag{35}$$

where squaring is valid because $\eta_- - \eta_+ > 0$. Multiplying by $\|\widehat{\mathbf{v}}_\star\|_2$, taking the minimum over j, and substituting $\eta_+^2 = 2(1-s)$ and $\eta_-^2 = 2(1-\kappa)$ yields the bound $\omega_\star \ge \|\widehat{\mathbf{v}}_\star\|_2\big[(1-\kappa) - 2\sqrt{(1-\kappa)(1-s)}\big] > 0$. □

Lemma 6.4. *The attended margin satisfies $\beta \ge \alpha_\star\omega_\star + (1-\alpha_\star)\omega_{\min}$. Hence $\beta > 0$ for any $\alpha_\star > 0$ if $\omega_{\min} \ge 0$, and for $\alpha_\star > -\omega_{\min}/(\omega_\star - \omega_{\min})$ if $\omega_{\min} < 0$.*

Proof of Lemma 6.4. Fix a candidate negative $\mathbf{z}_j^-$. By the definition of the attended memory direction,

$$\overline{\mathbf{v}}^\top(\mathbf{z}^+ - \mathbf{z}_j^-) = \alpha_\star \widehat{\mathbf{v}}_\star^\top(\mathbf{z}^+ - \mathbf{z}_j^-) + \sum_{k\ne\star} \alpha_k \widehat{\mathbf{v}}_k^\top(\mathbf{z}^+ - \mathbf{z}_j^-) \tag{36}$$

$$\ge \alpha_\star\omega_\star + (1-\alpha_\star)\omega_{\min}. \tag{37}$$

The inequality holds because the useful attended memory context has margin at least $\omega_\star$ against every candidate negative, each remaining context has margin at least $\omega_{\min}$, and the other attention weights

sum to $1 - \alpha_\star$. This lower bound holds for every j; taking the minimum over negatives proves the stated bound $\beta \geq \alpha_\star \omega_\star + (1 - \alpha_\star)\omega_{\min}$.

When $\omega_{\min} < 0$, the lower bound is positive precisely when

$$\alpha_\star \omega_\star + (1 - \alpha_\star)\omega_{\min} > 0 \tag{38}$$

$$\iff \qquad \alpha_\star > \frac{-\omega_{\min}}{\omega_\star - \omega_{\min}}. \tag{39}$$

When $\omega_{\min} \geq 0$, the lower bound is strictly positive for every $\alpha_\star > 0$, because $\omega_\star > 0$. □

As before, the attention mass required by Lemma 6.4 follows from a logit gap.

Lemma B.5. *If $K > 1$, $\omega_{\min} < 0$, and $\ell_\star - \ell_k \geq \delta$ for all $k \neq \star$, then $\beta > 0$ whenever $\delta > \tau_{\text{att}} \log\big((K-1)(-\omega_{\min})/\omega_\star\big)$.*

Proof of Lemma B.5. The logit-gap bound in Lemma B.2 gives

$$\alpha_\star \geq \frac{1}{1 + (K-1)\exp(-\delta/\tau_{\text{att}})}. \tag{40}$$

By Lemma 6.4, it suffices for this lower bound to exceed $-\omega_{\min}/(\omega_\star - \omega_{\min})$. Rearranging this inequality gives

$$(K-1)\exp(-\delta/\tau_{\text{att}}) < \frac{\omega_\star}{-\omega_{\min}}, \tag{41}$$

which is equivalent to $\delta > \tau_{\text{att}} \log\big((K-1)(-\omega_{\min})/\omega_\star\big)$. The resulting attention mass makes the lower bound of Lemma 6.4 strictly positive, and hence $\beta > 0$. □

B.4 InfoNCE Guarantees

We first prove the standard property used in the main text: for every j, $\widetilde{\mathbf{z}}_q^\top(\mathbf{z}^+ - \mathbf{z}_j^-) \geq -\tau \log\big(\exp(\mathcal{L}_\lambda) - 1\big)$, so $\mathcal{L}_\lambda < \log 2$ implies that $\widetilde{\mathbf{z}}_q$ ranks d^+ above all R negatives.

Proof. For each negative $\mathbf{z}_j^-$, define the margin of the augmented query as

$$\Delta_{\lambda,j} = \widetilde{\mathbf{z}}_q^\top(\mathbf{z}^+ - \mathbf{z}_j^-). \tag{42}$$

Substituting Eq. 42 into Eq. 4 gives

$$\mathcal{L}_\lambda = \log\left(1 + \sum_{k=1}^{R} \exp\left(-\frac{\Delta_{\lambda,k}}{\tau}\right)\right). \tag{43}$$

Exponentiating both sides of Eq. 43 yields

$$\exp(\mathcal{L}_\lambda) = 1 + \sum_{k=1}^{R} \exp\left(-\frac{\Delta_{\lambda,k}}{\tau}\right). \tag{44}$$

Therefore,

$$\sum_{k=1}^{R} \exp\left(-\frac{\Delta_{\lambda,k}}{\tau}\right) = \exp(\mathcal{L}_\lambda) - 1. \tag{45}$$

Since every term on the left-hand side of Eq. 45 is nonnegative, each individual term is at most their sum. Thus, for every j,

$$\exp\left(-\frac{\Delta_{\lambda,j}}{\tau}\right) \leq \exp(\mathcal{L}_\lambda) - 1. \tag{46}$$

Applying the increasing logarithm function to both sides gives

$$-\frac{\Delta_{\lambda,j}}{\tau} \leq \log(\exp(\mathcal{L}_\lambda) - 1). \tag{47}$$

Finally, multiplying by $-\tau$ reverses the inequality because $\tau > 0$:

$$\Delta_{\lambda,j} \geq -\tau \log(\exp(\mathcal{L}_\lambda) - 1)\,. \tag{48}$$

If $\mathcal{L}_\lambda < \log 2$, then $\exp(\mathcal{L}_\lambda) - 1 < 1$. Its logarithm is therefore negative, so the right-hand side is positive. Hence, $\Delta_{\lambda,j} > 0$ for every j, or equivalently, $\widetilde{\mathbf{z}}_q^\top \mathbf{z}^+ > \widetilde{\mathbf{z}}_q^\top \mathbf{z}_j^-$ for every candidate negative. □

Theorem 6.5 (Sufficient interpolation weight for frozen-query repair)**.** *Suppose the memory is discriminative, $\beta > 0$, and let $\mu_\lambda = \lambda\beta - (1-\lambda)\rho$. If $\lambda > \rho/(\rho+\beta)$, then $\widetilde{\mathbf{z}}_q^\top(\mathbf{z}^+ - \mathbf{z}_j^-) \geq \mu_\lambda > 0$ for all j, so $\widetilde{\mathbf{z}}_q$ ranks d^+ above every negative.*

Proof of Theorem 6.5. Fix an arbitrary candidate negative $\mathbf{z}_j^-$, and write $\mathbf{y}_\lambda = (1-\lambda)\mathbf{z}_q + \lambda\overline{\mathbf{v}}$ for the numerator in Eq. 3, so $\widetilde{\mathbf{z}}_q = \mathbf{y}_\lambda / \|\mathbf{y}_\lambda\|_2$. By linearity of the inner product,

$$\begin{aligned}
\mathbf{y}_\lambda^\top(\mathbf{z}^+ - \mathbf{z}_j^-) &= \big((1-\lambda)\mathbf{z}_q + \lambda\overline{\mathbf{v}}\big)^\top(\mathbf{z}^+ - \mathbf{z}_j^-) \\
&= (1-\lambda)\mathbf{z}_q^\top(\mathbf{z}^+ - \mathbf{z}_j^-) + \lambda\overline{\mathbf{v}}^\top(\mathbf{z}^+ - \mathbf{z}_j^-) \\
&= (1-\lambda)\gamma_j + \lambda\overline{\mathbf{v}}^\top(\mathbf{z}^+ - \mathbf{z}_j^-).
\end{aligned} \tag{49}$$

By definition, $\rho = \max_j[-\gamma_j]_+$, so $\gamma_j \geq -\rho$ for every j. By the definition of β, $\overline{\mathbf{v}}^\top(\mathbf{z}^+ - \mathbf{z}_j^-) = \beta_j \geq \beta$. Applying both lower bounds to Eq. 49 gives

$$\begin{aligned}
\mathbf{y}_\lambda^\top(\mathbf{z}^+ - \mathbf{z}_j^-) &\geq (1-\lambda)(-\rho) + \lambda\beta \\
&= \lambda\beta - (1-\lambda)\rho \\
&= \mu_\lambda.
\end{aligned} \tag{50}$$

The condition $\lambda > \rho/(\rho+\beta)$ is equivalent to

$$\lambda(\rho+\beta) > \rho, \tag{51}$$

which can be rearranged as

$$\lambda\beta > (1-\lambda)\rho. \tag{52}$$

Therefore,

$$\mu_\lambda = \lambda\beta - (1-\lambda)\rho > 0. \tag{53}$$

By Lemma 6.1, $\|\overline{\mathbf{v}}\|_2 \leq 1$. Since $\|\mathbf{z}_q\|_2 = 1$ and $\lambda \in (0,1)$, the triangle inequality gives

$$\begin{aligned}
\|\mathbf{y}_\lambda\|_2 &= \|(1-\lambda)\mathbf{z}_q + \lambda\overline{\mathbf{v}}\|_2 \\
&\leq (1-\lambda)\|\mathbf{z}_q\|_2 + \lambda\|\overline{\mathbf{v}}\|_2 \\
&\leq (1-\lambda) + \lambda \\
&= 1.
\end{aligned} \tag{54}$$

Because the unnormalized margin in Eq. 50 is positive (so, in particular, $\mathbf{y}_\lambda \neq \mathbf{0}$), normalization can only increase it:

$$\begin{aligned}
\widetilde{\mathbf{z}}_q^\top(\mathbf{z}^+ - \mathbf{z}_j^-) &= \frac{\mathbf{y}_\lambda^\top(\mathbf{z}^+ - \mathbf{z}_j^-)}{\|\mathbf{y}_\lambda\|_2} \\
&\geq \mathbf{y}_\lambda^\top(\mathbf{z}^+ - \mathbf{z}_j^-) \\
&\geq \mu_\lambda \\
&> 0.
\end{aligned} \tag{55}$$

Thus, the augmented query ranks the positive document above every candidate negative. □

Corollary 6.6 (InfoNCE after frozen-query repair)**.** *Under Theorem 6.5, if $\lambda > \rho/(\rho+\beta)$, then $\mathcal{L}_\lambda \leq \log\big(1 + R\exp(-\mu_\lambda/\tau)\big)$. This bound is below $\log 2$ when $\mu_\lambda > \tau \log R$, i.e., $\lambda > (\rho + \tau\log R)/(\rho+\beta)$.*

Proof of Corollary 6.6. Substituting the repaired margin $\widetilde{\mathbf{z}}_q^\top(\mathbf{z}^+ - \mathbf{z}_j^-) \geq \mu_\lambda$ into the InfoNCE loss in Eq. 4 gives

$$\begin{aligned}\mathcal{L}_\lambda &= \log\left(1 + \sum_{j=1}^{R} \exp\left(-\frac{\widetilde{\mathbf{z}}_q^\top(\mathbf{z}^+ - \mathbf{z}_j^-)}{\tau}\right)\right) \\ &\leq \log\left(1 + \sum_{j=1}^{R} \exp\Big(-\frac{\mu_\lambda}{\tau}\Big)\right) \\ &= \log\Big(1 + R\exp\Big(-\frac{\mu_\lambda}{\tau}\Big)\Big). \end{aligned} \tag{56}$$

If $\mu_\lambda > \tau \log R$, then $R\exp(-\mu_\lambda/\tau) < 1$, so the right-hand side is below $\log 2$. Finally, using $\mu_\lambda = \lambda(\rho+\beta) - \rho$ gives the equivalent condition $\lambda > (\rho + \tau\log R)/(\rho+\beta)$. □

B.5 Frozen-query repair through a Transformer

Theorem 6.7 (Frozen-query repair through a Transformer)**.** *If each map $\mathbf{E} \mapsto f_{\theta_q}(\mathbf{E})^\top(\mathbf{z}^+ - \mathbf{z}_j^-)$ has an H-Lipschitz gradient, then $\gamma_j(\lambda) \geq -\rho + \lambda\beta - \frac{1}{2}H\lambda^2\|\mathbf{\Delta}_q\|_F^2$ for all j and $\lambda \in [0,1]$. Hence the augmented query ranks d^+ above every negative if $\beta > \rho + \frac{1}{2}H\|\mathbf{\Delta}_q\|_F^2$.*

Recall the setup of Section 6: the reference input $\widetilde{\mathbf{E}}_{q,0}$ reproduces the frozen query, $f_{\theta_q}(\widetilde{\mathbf{E}}_{q,0}) = \mathbf{z}_q$; the path is $\widetilde{\mathbf{E}}_q(\lambda) = \widetilde{\mathbf{E}}_{q,0} + \lambda\mathbf{\Delta}_q$; the margins are $\gamma_j(\lambda) = f_{\theta_q}(\widetilde{\mathbf{E}}_q(\lambda))^\top(\mathbf{z}^+ - \mathbf{z}_j^-)$; and the memory gain is $\beta = \min_j \gamma_j'(0)$.

Proof of Theorem 6.7. The idea is to view each margin as a scalar function of the single variable λ and to bound how far it can fall below its tangent line at $\lambda = 0$. Fix a negative $j \in \{1,\dots,R\}$ and write $\mathbf{w}_j = \mathbf{z}^+ - \mathbf{z}_j^-$. The document embeddings come from the frozen document tower, so $\mathbf{w}_j$ does not depend on λ; only the query side moves.

To work with ordinary gradients, we vectorize the encoder input. Let $\mathbf{x} = \mathrm{vec}(\widetilde{\mathbf{E}}_q) \in \mathbb{R}^n$ with $n = Ld$, $\mathbf{x}_0 = \mathrm{vec}(\widetilde{\mathbf{E}}_{q,0})$, and $\Delta\mathbf{x} = \mathrm{vec}(\mathbf{\Delta}_q)$. The insertion path becomes the line segment $\mathbf{x}(\lambda) = \mathbf{x}_0 + \lambda\Delta\mathbf{x}$, and $\|\Delta\mathbf{x}\|_2 = \|\mathbf{\Delta}_q\|_F$ because vectorization preserves the Frobenius norm. With the margin map

$$F(\mathbf{x}) = f_{\theta_q}\big(\,\mathrm{unvec}(\mathbf{x})\big), \qquad \Gamma_j(\mathbf{x}) = F(\mathbf{x})^\top\mathbf{w}_j,$$

we have $\gamma_j(\lambda) = \Gamma_j(\mathbf{x}(\lambda))$, and it remains to lower-bound this one-dimensional function on $[0,1]$.

Its slope follows from the chain rule. Because $\mathbf{w}_j$ is fixed, $\nabla_{\mathbf{x}}\Gamma_j(\mathbf{x}) = J_F(\mathbf{x})^\top\mathbf{w}_j$, and because the path moves with constant velocity $\Delta\mathbf{x}$,

$$\gamma_j'(\lambda) = \big\langle \nabla_{\mathbf{x}}\Gamma_j(\mathbf{x}(\lambda)),\, \Delta\mathbf{x}\big\rangle = \big\langle J_F(\mathbf{x}(\lambda))\,\Delta\mathbf{x},\ \mathbf{w}_j\big\rangle. \tag{57}$$

At $\lambda = 0$, the vector $J_F(\mathbf{x}_0)\Delta\mathbf{x} = Df_{\theta_q}(\widetilde{\mathbf{E}}_{q,0})[\mathbf{\Delta}_q]$ is the direction in which the query embedding moves when memory is switched on infinitesimally, and its inner product with $\mathbf{w}_j$ is the resulting change in the margin of d^+ over d_j^-. This is exactly the slope used to define β, so $\gamma_j'(0) \geq \beta$.

The slope need not keep this initial value, and the smoothness assumption is what controls its drift. Since $\nabla_{\mathbf{x}}\Gamma_j$ is H-Lipschitz along the path, every $s \in [0,1]$ satisfies $\|\nabla_{\mathbf{x}}\Gamma_j(\mathbf{x}(s)) - \nabla_{\mathbf{x}}\Gamma_j(\mathbf{x}_0)\|_2 \leq H\|\mathbf{x}(s) - \mathbf{x}_0\|_2 = Hs\|\Delta\mathbf{x}\|_2$. Combining this with the first form of Eq. 57 and the Cauchy–Schwarz inequality gives

$$\big|\gamma_j'(s) - \gamma_j'(0)\big| = \big|\big\langle \nabla_{\mathbf{x}}\Gamma_j(\mathbf{x}(s)) - \nabla_{\mathbf{x}}\Gamma_j(\mathbf{x}_0),\, \Delta\mathbf{x}\big\rangle\big| \leq Hs\,\|\Delta\mathbf{x}\|_2^2,$$

so the slope can decrease at most linearly along the path, $\gamma_j'(s) \geq \gamma_j'(0) - Hs\,\|\Delta\mathbf{x}\|_2^2$.

Integrating this bound on the slope recovers a bound on the margin itself. A Lipschitz gradient is continuous, so γ_j is continuously differentiable and the fundamental theorem of calculus gives, for every $\lambda \in [0,1]$,

$$\gamma_j(\lambda) = \gamma_j(0) + \int_0^\lambda \gamma_j'(s)\,ds \geq \gamma_j(0) + \int_0^\lambda \big[\gamma_j'(0) - Hs\,\|\Delta\mathbf{x}\|_2^2\big]\,ds = \gamma_j(0) + \lambda\gamma_j'(0) - \tfrac{1}{2}H\lambda^2\|\Delta\mathbf{x}\|_2^2.$$

The true margin therefore lies above its tangent line at $\lambda = 0$, minus a quadratic correction.

It remains to express the starting point and the slope through the deficit and the gain. The reference input reproduces the frozen query, so $\gamma_j(0) = \mathbf{z}_q^\top \mathbf{w}_j$, and by definition of the deficit $-\gamma_j(0) \leq [-\gamma_j(0)]_+ \leq \rho$, i.e., $\gamma_j(0) \geq -\rho$. Substituting this together with $\gamma_j'(0) \geq \beta$, $\lambda \geq 0$, and $\|\Delta\mathbf{x}\|_2 = \|\mathbf{\Delta}_q\|_F$ yields

$$\gamma_j(\lambda) \geq -\rho + \lambda\beta - \tfrac{1}{2}H\lambda^2\|\mathbf{\Delta}_q\|_F^2.$$

Since j was arbitrary, this bound holds for every negative. At the full insertion $\lambda = 1$ it reads $\gamma_j(1) \geq \beta - \rho - \frac{1}{2}H\|\mathbf{\Delta}_q\|_F^2$, which is strictly positive whenever $\beta > \rho + \frac{1}{2}H\|\mathbf{\Delta}_q\|_F^2$. As $\gamma_j(1) = \widetilde{\mathbf{z}}_q^\top \mathbf{z}^+ - \widetilde{\mathbf{z}}_q^\top \mathbf{z}_j^-$, the augmented query is then strictly more similar to d^+ than to every negative d_j^-. □

The bound makes the role of each of its three terms transparent. The deficit $-\rho$ is fixed by the frozen retriever, the gain $\lambda\beta$ is what memory contributes, and the penalty $-\frac{1}{2}H\lambda^2\|\mathbf{\Delta}_q\|_F^2$ is the price of nonlinearity: it vanishes when $H = 0$ and grows quadratically with the size of the insertion. Because f_{θ_q} is ℓ_2-normalized, $H = 0$ is not expected in practice, even for the linear model. Three consequences follow. Discrimination is necessary for the certificate: if $\beta \leq 0$, the bound is at most $-\rho \leq 0$ and certifies nothing, because smoothness limits how far the encoder can drift but cannot create a useful direction. Larger insertions are not always better: for $\beta > 0$ and $H > 0$, the bound is maximized over $\lambda \in [0, 1]$ at $\lambda^\star = \min\{1, \beta/(H\|\mathbf{\Delta}_q\|_F^2)\}$, so a strong but curved insertion may be certified only once it is scaled down. Finally, the no-op assumption can be relaxed: the proof uses the reference input only through $\gamma_j(0) \geq -\rho$, so if the reference input does not reproduce $\mathbf{z}_q$ exactly, the argument holds verbatim with ρ replaced by the reference input's own deficit $\rho_0 = \max_j[-\gamma_j(0)]_+$.

C BASELINES

For every baseline, we maintain the same deployment constraint as REMEDI: the document encoder and the precomputed index stay fixed, so no method may re-encode the corpus.

C.1 QUERY-TOWER FINE-TUNING

These baselines ignore memory and instead adapt the query tower directly on the training queries, using the same InfoNCE objective, batch size, schedule, and early-stopping rule as A_ψ (Section 7).

- **Full FT** updates every query-tower parameter (learning rate 10^{-5}).
- **LoRA** (Hu et al., 2022) freezes the query tower and learns low-rank weight updates of rank 8, 16, or 32 (learning rate 4×10^{-5}).

Both leave the document tower untouched, so the index remains valid, but both change how *every* query is encoded, including queries that match no concept.

C.2 TRAINING-FREE MEMORY

These methods add memory without learning any parameters and the query encoder is used as is.

VectorPRF. Vector pseudo-relevance feedback (Li et al., 2023) enriches a query with the documents it already retrieves. The frozen query $\mathbf{z}_q$ first searches the corpus index to fetch the stored vectors of its top five documents $\mathbf{z}_{d_1}, \ldots, \mathbf{z}_{d_5}$ are averaged into it,

$$\mathbf{z}_q' = \mathrm{norm}\left(\mathbf{z}_q + \sum_{i=1}^{5} \mathbf{z}_{d_i}\right),$$

and $\mathbf{z}_q'$ performs a second search. This is the uniform-average rule of the original work. VectorPRF uses no concept memory and learns nothing, but it needs two corpus searches per query.

LexicalPRF. LexicalPRF is identical to VectorPRF except for the first retrieval round. When a query contains a rare concept, its dense embedding may itself be unreliable, so feedback documents retrieved with it may be off-topic. LexicalPRF therefore retrieves the top five feedback documents with BM25 instead, and then averages their stored dense vectors into $\mathbf{z}_q$ exactly as above.

C.3 Learned memory interfaces

These methods train a new memory interface, either alone on top of the frozen query encoder or jointly with query-side LoRA-r16 ($\alpha = 32$, dropout 0.1, on the attention Q/K/V/output projections). In both settings the document encoder, the memory encoders, and the original backbone weights stay frozen, and every interface is trained from scratch.

KPR-BERT. KPR (Yamada et al., 2025) is the baseline closest to REMEDI. It draws knowledge about each entity from external text such as Wikipedia and turns each passage into a memory vector via masked language modeling. One mention of the entity is masked, and the hidden state at the mask serves as the memory vector. A cross-attention layer at the output of the query tower then fuses these memory vectors into the query embedding. KPR differs from REMEDI in three ways. First, it augments only entities, whereas REMEDI can augment any query span that benefits from external memory. Second, it always appends a no-op slot to the fusion layer. Third, it fine-tunes the query-tower parameters jointly with the fusion layer. We attribute the severe regression we observe on out-of-domain benchmarks such as BEIR to this last choice.

MoMA. MoMA (Ge et al., 2023) is designed *specifically* for encoder-decoder retrievers and fuses memory at the encoder outputs. Its memory is built per query from a union of sources such as Wikipedia and PubMed: given a query, MoMA searches this corpus for relevant passages, fuses their embeddings with the query-encoder outputs, and passes the result to the decoder. The encoders we study (e.g., RoBERTa and Qwen) are encoder-only, so to apply MoMA we additionally train a two-layer Transformer decoder. This gives MoMA an advantage over the other baselines, since it uses these extra Transformer parameters.

D Experimental Details

We first describe the datasets, training and evaluation protocol, and concept-memory construction. The following section presents additional comparisons, design analyses, and qualitative examples.

D.1 Datasets, Encoders, and Evaluation

Table 6: Dataset statistics.

Dataset	Domain	Train	Val	Test	Corpus
MedQuAD	Consumer health	10K	1K	5K	16.4K
NQ	Open-domain QA	55.9K	2.9K	3.6K	21.0M
TriviaQA	Open-domain QA	57.4K	3.0K	11.3K	21.0M
EntityQ.	Open-domain QA	90.8K	11.4K	22.1K	21.0M
HotpotQA	Multi-hop QA	65.2K	3.4K	5.6K	88.3K
SQuAD	Reading comp.	83.2K	4.4K	10.6K	21.0K
PopQA	Long-tail QA	6.5K	0.8K	0.8K	140.9K
FEVER	Fact checking	67.7K	3.6K	10.4K	8.4K

Table 7: Encoder statistics.

Encoder	Layers	Dim	Params
RoBERTa-base	12	768	125M
E5-base-unsup.	12	768	109M
E5-base	12	768	109M
Nomic Embed v1	12	768	137M
Qwen3-Emb.-0.6B	28	1024	596M

Retrieval collections. The eight source datasets and their train, validation, test, and corpus sizes are summarized in Table 6. Natural Questions (NQ), TriviaQA, and EntityQuestions use the same

21,015,324-passage DPR Wikipedia corpus; HotpotQA, SQuAD, PopQA, and FEVER use their task-specific corpora. The five out-of-domain BEIR collections are SciFact, NFCorpus, FiQA-2018, ArguAna, and TREC-COVID. Each method searches the unchanged document index of its backbone. In particular, query LoRA and full query fine-tuning never update the document encoder or re-encode the corpus. Baseline architectures and their fixed-index adaptations are described in Appendix C.

Optimization and candidates. Trainable methods optimize query-to-document mean-positive InfoNCE over normalized embeddings, with temperature 0.05 and batches of 64 distinct queries. Documents supplied by other queries are in-batch negatives unless also positive for the current query.

We use AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$), weight decay 0.01 on matrices and zero decay on biases, normalization parameters, and learned boundary vectors. The learning rate is 10^{-4} for the memory adapter, 4×10^{-5} for query-side LoRA, and 10^{-5} for full query fine-tuning. A linear warm-up occupies 10% of updates, followed by cosine decay to zero; the global gradient norm is clipped at 1.0. With seed 42, training runs for at most 20 epochs, with a minimum of three, validation retrieval after each epoch, and early stopping with patience three. The adapter alone trains only its memory projections and boundary vectors; neither pretrained encoder is updated.

Full DPR search and relevance. For NQ, TriviaQA, and EntityQuestions, evaluation searches all 21,015,324 DPR Wikipedia passages, returning the top 100.

A DPR passage is relevant if its body contains any gold answer alias as a contiguous token sequence after Unicode NFD normalization and case-folding. With binary relevance r_i at rank i and R_q answer-containing passages in the *full corpus* (counted up to ten), the reported per-query score is

$$\mathrm{nDCG@10}(q) = \frac{\sum_{i=1}^{10} r_i / \log_2(i+1)}{\sum_{i=1}^{\min(10,R_q)} 1/\log_2(i+1)}.$$

A query with $R_q = 0$ scores zero. Other tasks use their supplied qrels. Reported nDCG@10 values are equal-weight test-query means multiplied by 100; help and harm are the percentages of queries with strictly higher and strictly lower per-query nDCG@10 than frozen, with ties in neither group.

D.2 Concept Memory and Activation

Extraction and matching. For non-MedQuAD queries, an LLM extracts exact contiguous spans of named or reusable concepts and returns lowercase, query-independent canonical keys for each. Matching a concept to a query is case-insensitive and word-bounded, permits whitespace/hyphen variation, and tries the extracted surface form, then the canonical name.

Memory construction. For non-MedQuAD datasets, each concept's memory contains five generated synopses followed by up to five BM25 passages. The synopsis generator sees only the canonical concept and a corpus description: it receives no query, answer, or retrieved document. It is asked for exactly five distinct, query-independent, one-to-three-sentence synopses covering different facts, subtopics, related entities, mechanisms, terminology, or aliases, returned in a JSON `synopses` list. We use GPT-4o with its default sampling configuration. The BM25 passages provide corpus evidence alongside these generated contexts.

For MedQuAD, we use the 4,741 `focus_area` labels as concepts and associate them with their 16,344 deduplicated answer documents. These user-provided contexts are constructed from all MedQuAD answers, including test answers, and can contain the evaluated query's exact gold answer; the controls in Appendix E.5 evaluate this dependence. In both constructions, the frozen document encoder encodes each context once and caches its representation, without changing the document index. Illustrative memory contents are collected separately in Appendix F.

Exact synopsis-generation templates. The following prompt wording is retained; the aliases line is omitted when none are available, and the kind line when the kind is absent or `other`.

System prompt.

> You write concise, query-independent background synopses of concepts for a `{corpus_description}`. Each synopsis captures a DIFFERENT angle of the concept — key facts, sub-topics, related entities and events, mechanisms, terminology, or aliases

Table 8: Test queries eligible for memory activation: a locatable concept span and a nonempty dictionary entry.

Dataset	Eligible	Test	Coverage (%)
MedQuAD	5,330	5,393	98.83
NQ	3,092	3,610	85.65
TriviaQA	4,693	11,313	41.48
EntityQuestions	16,985	22,075	76.94
HotpotQA	5,514	5,600	98.46
SQuAD	9,086	10,570	85.96
PopQA	799	814	98.16
FEVER	10,042	10,444	96.15

— so that a small frozen retrieval model can better understand the concept and match relevant documents. Describe only the concept itself; never refer to any specific question or answer. Keep each synopsis 1-3 sentences, dense with the terms a document about this concept would actually use.

User prompt.

Generate exactly 5 query-independent synopses for the concept below, as it would appear in a `{corpus_description}`. Each synopsis must capture a DIFFERENT angle or sub-topic; do not repeat the same facts. If the concept is ambiguous, commit to the sense implied by its canonical form.
concept: `{canonical_concept}`
aliases: `{comma-separated aliases}`
kind: `{concept kind}`
Return ONLY a JSON object:
`{"synopses": ["...", "...", ...]}`

Activation coverage. Table 8 counts test queries for which at least one concept has both a locatable span and a nonempty dictionary entry.

E Additional Experiments and Analysis

We examine adapter design choices and additional LoRA ranks, then analyze memory attention and compare alternative ways of using the same memory. A gold-answer removal control separates gains from access to the exact answer, and qualitative examples illustrate both improvements and failures.

E.1 RQ6: Design Choices

We ablate three design choices of the adapter A_ψ on MedQuAD for Qwen3 and RoBERTa (Table 9), changing one choice at a time from the reference configuration: pooled attention, two boundary tokens per side, and maximum contexts per concept. Token-conditioned attention (TokenAttn) reverses the order of pooling and attention. Pooled attention uses the frozen pooled query embedding as a single attention query. TokenAttn instead lets every query token embedding attend separately over the concept's context keys and values. It then mean-pools the attended outputs over the query tokens and projects the result into the inserted memory token.

(1) Pooled attention is better than token-conditioned attention. TokenAttn lowers nDCG@10 by 0.39 points on Qwen3 and by 3.01 on RoBERTa. A single query-level readout of memory is enough and is cheaper, since the attention is computed once per concept rather than once per token.

(2) Boundary tokens help the decoder-only encoder. Removing them costs Qwen3 0.72 points, while RoBERTa is unaffected ($+0.01$). One to four tokens per side perform within 0.3 points of the reference, so the exact number matters little.

(3) The weak encoder needs more memory. RoBERTa drops by 12.97 points with three contexts per concept and still by 1.15 with ten, because its parameters hold little of the required knowledge. Qwen3 saturates with three contexts (80.74), consistent with it needing memory only to fill gaps in its parametric knowledge.

Table 9: RQ6: adapter ablations on MedQuAD. Test query-macro nDCG@10 (%), difference from Ref. in points, and help/harm = % test queries scored higher/lower than frozen.

	Qwen3			RoBERTa		
Setting	nDCG@10	Δ	Help/Harm	nDCG@10	Δ	Help/Harm
Frozen	70.59	–	–	27.68	–	–
Ref. (PooledAttn, 2 per side, all)	80.57	–	30.3/6.5	**79.62**	–	69.6/3.3
Attention over contexts						
TokenAttn	80.18	−0.39	29.9/6.7	76.61	−3.01	68.4/3.5
Boundary tokens per side						
0	79.85	−0.72	29.5/6.8	**79.62**	+0.01	69.4/3.2
1	80.31	−0.27	30.1/7.1	79.45	−0.17	69.3/3.2
4	80.42	−0.15	29.9/6.3	79.59	−0.03	69.4/3.2
Memory Budget (per concept)						
3	**80.74**	+0.16	29.9/6.2	66.65	−12.97	57.9/5.4
5	80.38	−0.19	29.8/6.7	73.90	−5.72	64.2/4.1
10	80.40	−0.18	29.9/7.0	78.47	−1.15	68.2/3.7

Table 10: LoRA rank ablations on MedQuAD: query-macro test nDCG@10 (%) for ranks 4, 8, 16, and 32. Frozen, full query fine-tuning, and adapter-only REMEDI (PooledAttn) are reference methods. Mean, harm, and help average uniformly over five encoders; harm/help are the percentages of queries scoring lower/higher than the corresponding frozen encoder.

Method	RoBERTa	E5-unsup.	E5-base	Nomic	Qwen3	Mean	Harm	Help
Frozen	27.68	69.97	69.41	74.23	70.59	62.38	–	–
LoRA-r4	41.79	79.28	76.18	80.70	81.89	71.97	8.20	28.76
LoRA-r8	43.43	79.91	77.21	81.06	83.00	72.92	8.17	30.09
LoRA-r16	45.22	79.38	77.68	81.26	83.22	73.35	8.21	30.55
LoRA-r32	46.57	80.40	78.18	**81.30**	**83.32**	73.95	8.15	31.37
Full FT	54.04	**80.86**	**78.86**	81.26	82.71	75.55	8.08	33.32
REMEDI (PooledAttn)	**79.62**	80.48	74.89	80.81	80.57	**79.28**	**7.17**	**35.34**

E.2 LoRA Rank Ablations on MedQuAD

Table 10 ablates query-side LoRA rank over $r \in \{4, 8, 16, 32\}$ across five encoders, with Frozen, full query fine-tuning, and adapter-only REMEDI as reference methods. Increasing rank raises the five-encoder mean nDCG@10 from 71.97 at rank 4 to 73.95 at rank 32, with smaller gains at higher ranks. PooledAttn denotes the pooled-attention implementation of REMEDI *without* query-side LoRA; its mean is 79.28 with user-provided answer contexts that can contain test gold, compared with 75.55 for full query fine-tuning.

E.3 Attention Analysis

Does the adapter attend to the memory that is relevant to the query? For each activated concept, i.e., a concept matched in an evaluated query, let a be the adapter's attention over the concept's n memory slots. We compare a with a *relevance distribution* $s_i \propto \exp(\mathrm{sim}_i/\tau)$, where sim_i is the dot product between the L2-normalized frozen embeddings of slot i and of the query's gold passage (the most similar one when there are several), and $\tau = 0.05$ is the training temperature. This dot product is the score the frozen index uses for retrieval, so s is the frozen retriever's belief about which slot matches the answer. Our alignment metric is the *relevance KL ratio*

$$\rho = \frac{\sum D(a \,\|\, s)}{\sum D(u \,\|\, s)},$$

where D is the KL divergence, u is uniform attention over the n slots, and both sums run over activated concepts. $\rho = 0$ means that attention matches the relevance distribution, $\rho = 1$ that it is

Table 11: Attention analysis. $\hat{H}$: normalized attention entropy (1 = uniform). $\alpha_{\max}$: largest attention weight (uniform: .10 on NQ). ρ: relevance KL ratio $\sum D(a\|s)/\sum D(u\|s)$ (0 = attention matches relevance; 1 = no closer than uniform attention; > 1 = farther). Clean: % of attention on clean slots; uniform attention gives $10\times$ the number of clean slots, and the value is fixed (–) when all slots are clean or all are noise. Δ: nDCG@10 gain over frozen in points, from Table 4 (NQ) and Table 5 (MedQuAD). (a) NQ with adapters trained on each memory condition; 8,777 activated query–concept pairs per row. (b) MedQuAD concepts held out from training (Section 7.5); 1,393 test queries whose concept has at least two slots.

Setting	$\hat{H}$	$\alpha_{\max}$	ρ	Clean (%)	Δ
(a) NQ, Qwen3					
Clean (10/10 clean)	.96	.19	.94	–	+1.20
20% noise (8/10 clean)	.93	.23	.90	88	+0.27
50% noise (5/10 clean)	.81	.33	.73	79	−0.62
80% noise (2/10 clean)	.15	.91	.37	93	−0.12
All noise (0/10 clean)	1.00	.10	1.01	–	−0.67
(a) NQ, RoBERTa					
Clean (10/10 clean)	.83	.31	.79	–	+21.82
20% noise (8/10 clean)	.83	.31	.68	92	+20.63
50% noise (5/10 clean)	.74	.37	.54	83	+19.53
80% noise (2/10 clean)	.52	.58	.42	77	+15.63
All noise (0/10 clean)	.60	.52	1.63	–	+3.02
(b) MedQuAD, unseen concepts					
RoBERTa	.43	.71	.28	–	+53.77
Nomic	.81	.42	.88	–	+6.34
Qwen3	.96	.28	.98	–	+11.00

no closer to it than uniform attention, and $\rho > 1$ that it concentrates on irrelevant slots. Dividing by the uniform divergence makes ρ comparable across encoders and memory conditions, whose similarity distributions differ in sharpness; our conclusions are unchanged for $\tau \in \{0.02, 0.1\}$. We also report two measures of concentration, averaged over concepts: the normalized entropy $\hat{H} = H(a)/\log n \in [0, 1]$, which is 1 for uniform attention, and the largest attention weight $\alpha_{\max}$, which is $1/n$ for uniform attention. Gold labels are used only for this analysis, never for training or inference.

We analyze two settings (Table 11). On NQ, each adapter is trained and evaluated with one memory condition of Table 4; every condition is measured on the same 8,777 activated query–concept pairs from 6,295 test queries, each with ten slots. On MedQuAD, we use the concept-held-out split of Section 7.5. Of its 1,640 test queries, 1,619 activate a concept that is absent from training; we analyze the 1,393 whose concept has at least two memory slots.

(1) Attention follows relevance. Whenever NQ memory contains clean contexts, attention is closer to the relevance distribution than uniform attention ($\rho < 1$) for both encoders (Table 11(a)). As noise grows, the adapter concentrates on the few clean slots: from clean memory to 80% noise, ρ falls from .94 to .37 for Qwen3 and from .79 to .42 for RoBERTa, while $\hat{H}$ drops and $\alpha_{\max}$ rises, up to .91 for Qwen3. At 80% noise, the two clean slots receive 93% and 77% of the attention, against 20% under uniform weighting (Figure 3). Part of this trend reflects the contrast between clean and random slots, since random passages lie far from the gold passage. On clean memory, where this contrast is absent, RoBERTa's attention is both sharper and better aligned than Qwen3's ($\hat{H} = .83$ versus .96; $\rho = .79$ versus .94), and its NQ gain from memory is also far larger ($+21.82$ versus $+1.20$ points).

(2) Concentration is not relevance. Without informative slots, alignment disappears. With all-noise memory, RoBERTa still concentrates its attention almost as sharply as at 80% noise ($\hat{H} = .60$, $\alpha_{\max} = .52$), but on slots unrelated to the answer: $\rho = 1.63$, so its attention is farther from the relevance distribution than uniform attention would be. Its gain over frozen falls from $+15.63$ at 80% noise to $+3.02$. Qwen3's attention instead becomes uniform ($\hat{H} = 1.00$, $\alpha_{\max} = .10$, $\rho = 1.01$). Entropy and maximum weight alone would thus mistake RoBERTa's all-noise attention for selective reading; ρ separates the two cases.

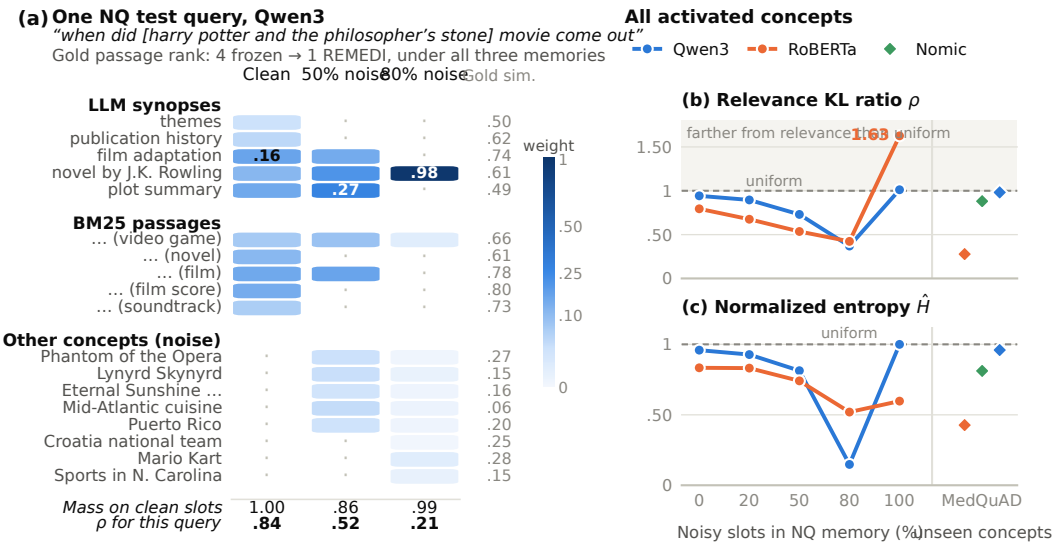


Figure 3: Attention on clean and noisy concept memory. (a) An illustrative NQ test query (Qwen3) under clean, 50%-noise, and 80%-noise memory. Attention concentrates on the clean synopses as noise increases; for this query, the gold passage remains at rank 1 (frozen: rank 4). (b) Mean attention on clean slots versus the share of noisy slots, providing a visual counterpart to Table 11(a). The gray line denotes the clean-slot share, i.e., the attention assigned by uniform weighting.

(3) On unseen concepts, alignment tracks reliance on memory. On held-out MedQuAD concepts, RoBERTa gains the most from memory (+53.77) and also attends most closely to the relevance distribution ($\rho = .28$), concentrating on few slots ($\hat{H} = .43$, $\alpha_{\max} = .71$; Table 11(b)). Its adapter therefore reads the contexts closest to the answer even for concepts it never saw in training. Nomic and Qwen3 spread attention more widely ($\hat{H} = .81$ and .96) and stay close to the uniform baseline ($\rho = .88$ and .98), yet still gain 6.34 and 11.00 points. For these stronger encoders, the gain appears to come from the concept's memory as a whole rather than from a single context. This is consistent with Section 7.4, where Qwen3 stays 8.4 points above frozen on MedQuAD even when all ten slots are replaced by other health answers.

E.4 Alternative Uses of Concept Memory

We compare two design variants using our same matched concept memory under the protocol of Table 1: REMEDI (Text) and REMEDI (MoMA reader). The latter adapts MoMA's reader architecture (Ge et al., 2023); we do not claim that architecture as our invention.

REMEDI (Text). REMEDI (Text) needs no training. After the query it appends one line, "Concept c: *text*", per memory item. The result is encoded with the frozen query encoder and its query prompt, truncated to the encoder's query length: 128 tokens for RoBERTa and 512 for Qwen3 and Nomic. On NQ, EntityQuestions, and TriviaQA, the items are the ten-item memory of every matched concept: five LLM synopses followed by up to five BM25 passages. On MedQuAD, they are the first ten answers of the query's focus-area concept, which may include the test query's gold answer. A query with no matched concept is left unchanged. Retrieval uses the same frozen index.

Table 12 shows that REMEDI (Text) has the highest EntityQuestions score for every encoder and improves all four RoBERTa sources. On Qwen3 and Nomic, however, it lowers MedQuAD by 14.78 and 22.17 points, NQ by 5.94 and 9.59, and TriviaQA by 2.00 and 2.99. Its overall mean, 31.02, is below Frozen (32.91) and REMEDI (38.36).

REMEDI (MoMA reader). REMEDI (MoMA reader) uses MoMA's reader (Ge et al., 2023), which attends jointly over query and evidence tokens. It takes the evidence from the matched concepts' memory instead of a learned augmentation retriever, which also removes MoMA's alternating training

Table 12: REMEDI (Text): test nDCG@10 (%) on the four source datasets. Query and document encoders and the document index are frozen.

	Qwen					RoBERTa					Nomic					Overall
Method	MQ	NQ	EQ	TQA	Mean	MQ	NQ	EQ	TQA	Mean	MQ	NQ	EQ	TQA	Mean	Mean
Frozen	70.59	29.02	29.07	38.89	41.89	27.68	6.60	4.00	8.28	11.64	74.23	35.25	28.74	42.55	45.19	32.91
REMEDI (Text)	55.81	23.08	**32.45**	**36.89**	37.06	34.52	**12.32**	**13.43**	**13.88**	18.54	52.06	25.66	**32.63**	39.56	37.48	31.02
REMEDI	**80.57**	**25.96**	26.64	36.46	**42.41**	**79.62**	11.34	6.07	12.51	**27.39**	**80.81**	**32.44**	27.42	**40.51**	**45.30**	**38.36**

Table 13: REMEDI (MoMA reader) versus REMEDI: test nDCG@10 (%) on the four source datasets and the five-task BEIR mean of the MedQuAD-trained checkpoint. The document encoder and index are frozen; + LoRA adds query-side LoRA-r16. **Bold** scores are the best non-Frozen scores per column.

	Qwen						RoBERTa						Nomic						Overall	
Method	MQ	NQ	EQ	TQA	Mean	BEIR	MQ	NQ	EQ	TQA	Mean	BEIR	MQ	NQ	EQ	TQA	Mean	BEIR	Mean	BEIR
Frozen	70.59	29.02	29.07	38.89	41.89	61.82	27.68	6.60	4.00	8.28	11.64	18.06	74.23	35.25	28.74	42.55	45.19	54.59	32.91	44.82
REMEDI (MoMA reader)	84.66	22.67	25.14	28.93	40.35	56.09	80.87	11.59	8.32	12.11	28.22	16.29	85.65	30.37	24.45	35.34	43.96	48.31	37.51	40.23
REMEDI (MoMA reader) + LoRA	**85.16**	24.67	24.70	28.60	40.78	52.40	**82.30**	12.66	**9.11**	13.21	29.32	8.88	**85.83**	29.35	27.22	35.27	44.41	46.10	38.17	35.79
REMEDI	80.57	**25.96**	**26.64**	**36.46**	**42.41**	**61.59**	79.62	11.34	6.07	12.51	27.39	**17.42**	80.81	**32.44**	**27.42**	**40.51**	**45.30**	**54.19**	**38.36**	**44.40**
REMEDI + LoRA	83.34	21.57	23.88	31.19	40.00	59.54	81.93	**13.70**	8.90	**15.47**	**30.00**	14.67	81.37	31.15	26.53	36.16	43.80	51.14	37.93	41.78

loop. Its memory is the same ten items per concept on the open-domain tasks and all of the concept's answers on MedQuAD. Context token states are cached from the frozen encoder. A new two-layer Transformer decoder of backbone width reads the query and context states through a learned readout token, with learned embeddings marking query versus context tokens. A linear projection followed by ℓ_2 normalization maps the readout into the frozen document space. Queries with no matched concept keep their frozen encoding. We train the reader alone at learning rate 10^{-4}, and jointly with query-side LoRA-r16 at 4×10^{-5} for the LoRA weights. By contrast, the MoMA row of the main text retrieves its evidence by dense search over the union of all dictionary documents.

In Table 13, REMEDI (MoMA reader) + LoRA has the highest MedQuAD score for all three encoders: 85.16, 82.30, and 85.83. Even adapter-only, it exceeds REMEDI on MedQuAD by 4.09, 1.25, and 4.84 points on Qwen3, RoBERTa, and Nomic. Like the other variants, these user-provided contexts are built from all MedQuAD answers and can contain the test query's exact gold answer (Table 14). The gain does not carry over to the other evaluations. REMEDI (MoMA reader) without LoRA lowers BEIR by 5.73 and 6.28 points on Qwen3 and Nomic, compared with 0.23 and 0.40 for REMEDI, and it trails REMEDI on NQ and TriviaQA for both encoders. Across encoders, REMEDI has the highest source mean (38.36), against 37.51 for REMEDI (MoMA reader) and 38.17 with LoRA. It also has the highest BEIR mean among the trained methods (44.40), against 40.23 and 35.79.

E.5 Gold-Answer Memory Controls

The user-provided MedQuAD contexts associate each concept with all its answer documents, including test answers, so a test query can read its own gold document, encoded exactly as in the index. We compare four memory policies: all answers (reference), all except contexts whose normalized text equals the current gold, training answers only with the gold removed, and a gold-only oracle. The two removal policies are evaluated both on the reference checkpoint with its memory changed *only at test time* and after training a new adapter under the corresponding policy. The frozen index is identical in every condition. A query with an empty memory bag falls back to the frozen encoding: 4,607 and 4,458 of 5,393 queries stay active under gold removal and train-answers-only removal, respectively. No removed-gold bag retains the gold text, although 39 retain a near-duplicate (unigram Jaccard ≥ 0.8); texts shared verbatim by training and evaluation queries can remain under the train-only policy.

For Qwen3 the improvement is not contingent on storing the exact gold: removing it only at test time changes 80.57 to 79.30, and training and evaluating without it reaches 80.54 (80.04 when only training answers are eligible). For RoBERTa, in contrast, much of the reference gain depends on the gold. Test-only removal drops 79.62 below frozen to 21.86, whereas the gold-only oracle reaches 94.45. When retrained without the gold, RoBERTa still improves from 27.68 to 43.50, or 41.24 with training answers only: there is useful information beyond the gold, but the reference 79.62 should not be interpreted as a gold-free result. On the 4,607-query nonempty-memory subset, retraining without the gold improves on frozen by 11.65 points for Qwen3 and 18.52 for RoBERTa. The retrospective

Table 14: MedQuAD user-provided answer-context control: query-macro test nDCG@10 (%), change from Frozen (points), and percentages helped/harmed. Reference memory includes each test query's gold answer. Removing it at test time uses the reference checkpoint; the trained-and-tested rows retrain without it. The index is unchanged; empty bags revert to frozen.

	Qwen3			RoBERTa		
User-provided contexts	nDCG@10	Δ	Help/Harm	nDCG@10	Δ	Help/Harm
Frozen	70.59	–	–	27.68	–	–
All answers (including test gold)	80.57	+9.98	30.3/6.5	79.62	+51.93	69.6/3.3
Test time, reference checkpoint						
All answers, exact gold removed	79.30	+8.70	28.1/7.3	21.86	−5.82	18.2/24.4
Training answers, exact gold removed	78.83	+8.24	27.1/7.1	19.50	−8.19	13.6/24.3
Exact gold only (oracle)	81.43	+10.83	30.9/4.2	94.45	+66.77	78.0/0.1
Trained and tested under removal policy						
All answers, exact gold removed	80.54	+9.95	29.0/5.5	43.50	+15.82	35.7/8.5
Training answers, exact gold removed	80.04	+9.45	28.1/5.4	41.24	+13.56	32.7/8.5
Queries with nonempty gold-removed memory (4,607 queries)						
Frozen	66.79	–	–	23.30	–	–
All answers (including test gold)	78.15	+11.36	34.4/7.4	77.36	+54.06	73.0/3.9
All answers, exact gold removed (test time)	76.98	+10.19	32.9/8.5	16.48	−6.82	21.3/28.5
All answers, exact gold removed (trained)	78.43	+11.65	33.9/6.4	41.82	+18.52	41.8/10.0

unseen-concept results of Table 5 use the original all-answer user-provided contexts and do not by themselves remove this qualification.

E.6 Qualitative Examples

The following examples illustrate when concept memory helps retrieval and when it shifts retrieval toward less useful information. We show memory attention, gold-passage ranks, and representative retrieved passages. The retrieved top-10 lists were summarized using GPT-4o, given the query, concept, and gold passage, with the two systems presented anonymously in random order. These summaries describe passage content rather than MedQuAD metadata and are used to interpret the examples.

MedQuAD: retrieval improvements. Tables 15 and 16 show four cases where matched memory improves the gold rank, alongside wrong-concept memory controls. For the Qwen3 Glomerular Diseases and Hashimoto thyroiditis examples, wrong memory still lifts the gold to ranks 1 and 2, so the improvements cannot be attributed to the matched concept alone. For Nomic on Glomerular Diseases, by contrast, wrong memory leaves the gold at rank 82. The summaries also highlight a limitation of MedQuAD's single-gold metric: frozen retrieval can return useful alternatives, including near-duplicates of the Hashimoto thyroiditis and Primary Hyperparathyroidism gold answers.

MedQuAD: retrieval failures. Table 17 shows two Qwen3 cases where memory can overemphasize generic text. For "Is Friedreich ataxia inherited?", the adapter puts .68 attention on "This condition is inherited in an autosomal recessive pattern . . . ", a templated passage that does not name the disease, and .12 on the gold. It then retrieves inheritance passages for other conditions and loses the gold from the top 100; Nomic also changes from gold rank 1 to outside the top 100 on this query. For Simple Kidney Cysts, attention is diffuse (.11–.16), including .14 on "The kidneys are two bean-shaped organs . . . "; the new top results favor generic kidney anatomy and the gold falls to rank 29. Wrong memory keeps the gold at rank 1 or 2 in these cases, so the matched memory's content, not insertion alone, is implicated.

NQ: retrieval with concept-only memory. Table 18 shows one NQ test query under Qwen3, whose adapter is trained and evaluated with the clean ten-slot memory of Table 4. This memory is built from the concept name alone, from five LLM synopses and five BM25 passages, so unlike the all-answer MedQuAD user-provided contexts, it need not contain a gold passage; for this query it contains none. For "who is the little girl who played cindy lou who", the GPT-4o summary describes

Table 15: Glomerular Diseases examples improved by REMEDI under two encoders. Memory entries are shown by question type and attention weight. Gold ranks and the top three passages under frozen, matched, and wrong-concept memory are followed by GPT-4o summaries of the frozen and matched top-10 lists. ⋆ denotes the gold passage; GD: Glomerular Diseases.

(a) Qwen3: *"What is (are) Glomerular Diseases?"* Wrong memory: growth hormone deficiency

Memory (attention): definition⋆ .18, causes .17, diagnosis .17, definition .16, what to do .16, symptoms .16

Retriever	Gold rank	Top 3 retrieved (concept · question)
Frozen	50	GD · causes; GD · causes; GD · diagnosis
REMEDI	1	**GD · definition⋆**; GD · definition; High Blood Pressure and Kidney Disease · definition
Wrong	1	**GD · definition⋆**; GD · definition; Kidney Disease · causes

GPT-4o summary, frozen: "emphasizes causes, symptoms, and diagnosis of glomerular disease, alongside descriptions of specific glomerulopathies." *On* REMEDI*:* "places the gold passage first and an apparent duplicate second, then largely retrieves general kidney explanations rather than glomerular disease itself."

(b) Nomic: *"What is (are) Glomerular Diseases?"* Wrong memory: growth hormone deficiency

Memory (attention): diagnosis .18, definition .17, definition⋆ .17, symptoms .17, what to do .16, causes .15

Retriever	Gold rank	Top 3 retrieved (concept · question)
Frozen	>100	GD · causes; GD · causes; fibronectin glomerulopathy · definition
REMEDI	1	**GD · definition⋆**; GD · definition; GD · what to do
Wrong	82	GD · causes; GD · causes; GD · symptoms

GPT-4o summary, frozen: "retrieves causes, symptoms, testing, and overview passages about glomerular disease, alongside descriptions of specific kidney disorders." *On* REMEDI*:* "puts the gold passage first and its near-duplicate second, but also drifts into general kidney material."

the frozen results as performers known for other roles, several of them other characters named Cindy. The first gold passage is at rank 97. The memory covers the 1957 book, the 1966 television special, the 2000 and 2018 films, and the stage musical, and one synopsis states that Taylor Momsen played Cindy Lou Who in the 2000 film. REMEDI gives that synopsis only .14 of its attention, the fifth-largest of the ten weights, yet ranks both gold Taylor Momsen passages first and second, and a third Momsen passage, also identified by GPT-4o as answering the query, sixth. Under RoBERTa, frozen retrieval leaves both gold passages outside the top 100; REMEDI puts .22 of its attention on the same synopsis, its second-largest weight, and ranks a gold passage first. We chose this query to illustrate the mechanism: on NQ, clean memory raises Qwen3's nDCG@10 by only 1.20 points in aggregate (Table 4). The retrieval lists were summarized using GPT-4o as above, with both gold passages supplied.

Table 16: Two further MedQuAD examples improved by REMEDI, continuing Table 15 with the same conventions. PH: Primary Hyperparathyroidism.

(c) Qwen3: *"What is (are) Hashimoto thyroiditis?"* Wrong memory: progressive myoclonic epilepsy

Memory (attention): inheritance .27, genetics .20, prevalence .19, treatments .18, definition* .16

Retriever	Gold rank	Top 3 retrieved (concept · question)
Frozen	19	Hashimoto's disease · definition; Hashimoto's disease · what to do; hypothyroidism · causes
REMEDI	1	**Hashimoto thyroiditis · definition***; hypothyroidism · symptoms; Hashimoto's disease · symptoms
Wrong	2	Hashimoto's disease · definition; **Hashimoto thyroiditis · definition***; Hashimoto's syndrome · definition

GPT-4o summary, frozen: "omits the gold passage but retrieves several alternative descriptions of Hashimoto disease and passages about its causes, prevalence, inheritance, and management." *On* REMEDI*:* "places the gold definition first, then mixes other descriptions of Hashimoto disease with passages about its symptoms and related thyroid conditions."

(d) Qwen3: *"What is (are) Primary Hyperparathyroidism?"* Wrong memory: yeast infections

Memory (attention): 11 passages (four definitions, two each for diagnosis and what to do, and one each for causes, symptoms and treatments), weighted .08 to .11, with the largest weight on definition* (.11)

Retriever	Gold rank	Top 3 retrieved (concept · question)
Frozen	68	PH · what to do; PH · definition; PH · causes
REMEDI	1	**PH · definition***; Parathyroid Disorders · definition; PH · what to do
Wrong	46	PH · what to do; PH · causes; PH · definition

GPT-4o summary, frozen: "leads with two passages that directly define primary hyperparathyroidism, then covers its causes, diagnosis, symptoms, and diet." *On* REMEDI*:* "retrieves the gold passage at rank 1 but drifts into unrelated pituitary material and genetic causes of a related condition."

Table 17: Two MedQuAD test queries harmed by REMEDI, with the same conventions as Table 15. FA: Friedreich ataxia; SKC: Simple Kidney Cysts.

(a) Qwen3: *"Is Friedreich ataxia inherited?"* Wrong memory: dengue fever

Memory (attention): inheritance .68 (templated, concept not named), inheritance* .12, genetics .05, causes .03, prevalence .03, definition .03, treatments .02, definition .02, symptoms .02

Retriever	Gold rank	Top 3 retrieved (concept · question)
Frozen	1	**FA · inheritance***; FA · causes; FA · prevalence
REMEDI	>100	Perrault syndrome · inheritance; Rabson–Mendenhall syndrome · inheritance; North American Indian childhood cirrhosis · inheritance
Wrong	1	**FA · inheritance***; FA · prevalence; FA · causes

GPT-4o summary, frozen: "ranks the exact gold answer first and retrieves additional passages about Friedreich ataxia, some of which independently answer the yes-or-no question." *On* REMEDI*:* "All ten passages describe autosomal recessive inheritance and carriers for an unnamed condition. None identifies Friedreich ataxia, so the text does not establish that it answers the query."

(b) Qwen3: *"What is (are) Simple Kidney Cysts?"*Wrong memory: D-2-alpha hydroxyglutaric aciduria

Memory (attention): what to do .16, definition .14 (kidney anatomy), causes .12, diagnosis .12, treatments .12, definition* .12, what to do .11, symptoms .11

Retriever	Gold rank	Top 3 retrieved (concept · question)
Frozen	1	**SKC · definition***; SKC · what to do; SKC · causes
REMEDI	29	Prevent diabetes problems: Keep your kidneys healthy · definition; Solitary Kidney · definition; Growth Failure in Children with Chronic Kidney Disease · definition
Wrong	2	SKC · what to do; **SKC · definition***; SKC · causes

GPT-4o summary, frozen: "ranks the gold definition first and stays largely focused on simple kidney cysts." *On* REMEDI*:* "retrieves repetitive passages describing the kidneys' anatomy and how they filter blood and produce urine. None of its passages describes simple kidney cysts."

Table 18: An NQ example under Qwen3 with clean ten-slot memory. Memory entries are ordered by attention and shown with their source and an excerpt. The bold entry names the answer; no entry is a gold passage. Gold rank is the better rank of the two gold passages. The top three retrieved passages are followed by GPT-4o summaries of the top-10 lists. ⋆ denotes a gold passage.

Qwen3: *"who is the little girl who played cindy lou who"* Concept: cindy lou who

Attn.	Source	Memory entry (excerpt)
.21	Synopsis	In the 2018 computer-animated film "The Grinch" produced by Illumination, Cindy Lou Who is updated as a modern, energetic child …
.17	BM25	*Dr. Seuss' How the Grinch Stole Christmas! The Musical*: … Kayley Stallings and Issadora Ava Tulalian as Cindy Lou Who …
.14	Synopsis	Cindy Lou Who is a fictional Who child from Dr. Seuss's 1957 book "How the Grinch Stole Christmas!" …
.14	Synopsis	In the 1966 animated television special … Cindy Lou Who appears as a young, wide-eyed Who …
.14	Synopsis	**The 2000 live-action film adaptation … expands Cindy Lou Who into a central character portrayed by Taylor Momsen** …
.07	BM25	*The Grinch (film)*: … 6-year-old Cindy Lou Who notices that her mother Donna is overworked …
.04	Synopsis	Across various Dr. Seuss adaptations and licensed media, Cindy Lou Who has become a recurring cultural icon …
.03	BM25	*How the Grinch Stole Christmas (2000 film)*: … He then sees Cindy Lou on top of the sleigh …
.03	BM25	*How the Grinch Stole Christmas! (TV special)*: … almost caught by Cindy Lou Who (voiced by an uncredited June Foray) …
.03	BM25	*How the Grinch Stole Christmas (2000 film)*: … The mayor then shames Cindy Lou for inviting the Grinch …

Retriever	Gold rank	Top 3 retrieved (passage title)
Frozen	97	Lindsay and Sidney Greenbush (Carrie Ingalls in *Little House on the Prairie*); Rich Little; Cindy Carol
REMEDI	1	**Taylor Momsen**⋆; **Taylor Momsen**⋆; Natasha Melnick (Cindy Sanders in *Freaks and Geeks*)

GPT-4o summary, frozen: "mostly retrieves performers associated with other characters named Cindy or child-acting roles, alongside several Little House on the Prairie actors. These passages identify people who played or voiced different roles, not Cindy Lou Who." *On* REMEDI*:* "Beyond its two gold passages, [it] includes another Taylor Momsen passage that answers the query and a passage about Cindy Lou Who's film plot. Its other results describe actresses, roles, or casting for different characters and productions."

F MEMORY EXAMPLES

Table 19 gives an overview of illustrative concept memories. Tables 20 and 21 show stored entries from seven of these memories. On MedQuAD the entries are the concept's answer documents; on the other datasets they are GPT-4o synopses and BM25 passages. The banks are reusable but imperfect: alongside on-concept evidence they hold generic text, near-name neighbors, and different entities that share a name. Whether the reader uses the right entries is measured by the experiments, not by these examples.

Table 19: Illustrative concepts with saved memory. "Answers" denotes MedQuAD answer memory; the other examples have generated synopses and BM25 contexts. The final column describes the information or limitation illustrated, not a measured retrieval gain.

Dataset	Concept	Memory	Illustrative role
MedQuAD	Dystonias	Answers	Definition, treatment, and progression
MedQuAD	Rapid-onset dystonia parkinsonism	Answers	Clinical presentation, ATP1A3, and inheritance
MedQuAD	Glomerular Diseases	Answers	Symptoms, diagnosis, causes, and generic kidney text
MedQuAD	Hashimoto thyroiditis	Answers	Related disease names and multiple medical facets
MedQuAD	Friedreich ataxia	Answers	Named genetic evidence versus generic inheritance text
HotpotQA	Arthur's Magazine	Synopses + BM25	Publication chronology and overlapping source evidence
HotpotQA	First for Women	Synopses + BM25	Concept-specific synopses with off-target lexical hits
HotpotQA	Scott Derrickson	Synopses + BM25	Named-person descriptions and filmmaking terminology
HotpotQA	Ed Wood	Synopses + BM25	Ambiguity between a filmmaker and a film
NQ	Harry Potter and the Philosopher's Stone	Synopses + BM25	Novel, film, video game, and soundtrack senses
PopQA	John Mayne	Synopses + BM25	A poet in synopses versus other people in lexical hits
FEVER	Roman Atwood	Synopses + BM25	Relevant biographical content and surname collisions

Table 20: MedQuAD answer memories for three concepts, labeled by the facet each answer covers. Text is verbatim; … marks our truncation.

Facet	Memory entry (excerpt)
Concept: dystonias	
Definition	The dystonias are movement disorders in which sustained muscle contractions cause twisting and repetitive movements or abnormal postures. The movements, which are involuntary and sometimes painful, may affect a single muscle; a group of muscles such as those in the arms, legs, or neck; or the entire body. …
Treatment	No one treatment has been found to be universally effective. Instead, doctors use a variety of therapies (medications, surgery, and other treatments such as physical therapy, splinting, stress management, and biofeedback) aimed at reducing or eliminating muscle spasms and pain. …
Progression	… Dystonias often progress through various stages. Initially, dystonic movements are intermittent and appear only during voluntary movements or stress. Later, individuals may show dystonic postures and movements while walking and ultimately even while they are relaxed. …
The running example of Section 1. Treatment names and progression terminology come from separate documents, not from a repeated definition.	
Concept: rapid-onset dystonia parkinsonism	
Presentation	Rapid-onset dystonia parkinsonism is a rare movement disorder. "Rapid-onset" refers to the abrupt appearance of signs and symptoms over a period of hours to days. … Parkinsonism can include tremors, unusually slow movement (bradykinesia), rigidity, …
Mechanism	Rapid-onset dystonia parkinsonism is caused by mutations in the ATP1A3 gene. This gene provides instructions for making one part of a larger protein called Na^+/K^+ ATPase, also known as the sodium pump. …
Inheritance	This condition is inherited in an autosomal dominant pattern, which means one copy of the altered ATP1A3 gene in each cell is sufficient to cause the disorder. … Not everyone who has an ATP1A3 mutation will ultimately develop the signs and symptoms of rapid-onset dystonia parkinsonism. …
The memory links ATP1A3 to the sodium pump. The mechanism entry is also the gold answer to this concept's genetics query; Table 14 controls for such gold-bearing memory.	
Concept: friedreich ataxia	
Cause	What causes Friedreich ataxia? Friedreich ataxia is caused by mutations in the FXN gene. This gene provides instructions for making a protein called frataxin. One region of the FXN gene contains a segment of DNA known as a GAA trinucleotide repeat. …
Inheritance	How is Friedreich ataxia inherited? Friedreich ataxia is inherited in an autosomal recessive manner. This means that to be affected, a person must have a mutation in both copies of the responsible gene in each cell. …
Inheritance	This condition is inherited in an autosomal recessive pattern, which means both copies of the gene in each cell have mutations. The parents of an individual with an autosomal recessive condition each carry one copy of the mutated gene, but they typically do not show signs and symptoms of the condition.
The last entry never names the disease. A reader that emphasizes it keeps the inheritance topic but can lose the disease identity.	

Table 21: Synopsis and BM25 memories for four concepts. Synopses are generated from the concept name alone; BM25 passages are stored truncated and start with their title. Text is verbatim up to casing; … marks our truncation. †: the entry does not describe the concept.

Source	Memory entry (excerpt)
HotpotQA concept: arthur's magazine	
Synopsis	Arthur's Magazine was a 19th-century literary periodical published in Philadelphia, primarily focused on fiction, poetry, and essays.
Synopsis	The magazine was established in 1844 by Timothy Shay Arthur, a prominent American author and editor.
BM25	*Arthur's Magazine*: Arthur's Magazine (1844–1846) was an American literary periodical published in Philadelphia in the 19th century. Edited by Timothy Shay Arthur
BM25†	*Arthur's Lady's Home Magazine*: Arthur's Home Magazine (1852–ca. 1898) or Ladies' Home Magazine was an American periodical published in
The synopses and the first passage agree on period and place; the second passage is a different magazine with a similar title.	
HotpotQA concept: first for women	
Synopsis	First for Women is a biweekly lifestyle magazine catering primarily to middle-aged women, featuring content on health, nutrition, beauty, relationships, and inspirational stories.
Synopsis	In print since 1989, First for Women competes with similar women's magazines such as Woman's World and Good Housekeeping, distinguishing itself with concise, actionable content.
BM25†	*Rabotnitsa*: Rabotnitsa … is a women's journal, published in the Soviet Union and Russia and one of the oldest Russian
BM25†	*The American Jewess*: The American Jewess (1895–1899) described itself as "the only magazine in the world devoted to the interests of Jewish women."
Only the synopses describe the concept; the passages share topic words but concern other publications.	
PopQA concept: john mayne	
Synopsis	John Mayne was an 18th-century Scottish poet best known for his work 'The Siller Gun,' a poem describing the medieval custom of shooting competitions in Dumfries.
Synopsis	John Mayne also contributed to journalism, working for several years as an editor and writer for newspapers in both Scotland and London.
BM25	*John D. Mayne*: John Dawson Mayne QC (1828–1917) was a British lawyer and legal expert who served as acting
BM25	*John D. Mayne*: to many of his relatives. John Dawson Mayne QC (1828–1917) was a British lawyer and
The synopses and the passages describe two different people named John Mayne: more entries are not more useful evidence.	
FEVER concept: roman atwood	
Synopsis	Roman Atwood is a prominent American content creator best known for his YouTube channels, which feature vlogs, pranks, and family-focused content, especially under the channel 'Roman Atwood Vlogs.'
Synopsis	Atwood initially gained popularity for his elaborate prank videos, many of which went viral, helping to define his early career as a YouTuber and entertainer.
BM25	*Roman Atwood*: Roman Bernard Atwood (born May 28, 1983) is an American YouTube personality, comedian, vlogger, and prankster. He
BM25†	*Margaret Atwood*: Margaret Eleanor Atwood (born November 18, 1939) is a Canadian poet, novelist, literary critic, essayist, inventor, teacher,
An exact biographical match and a surname collision sit in the same bank.	